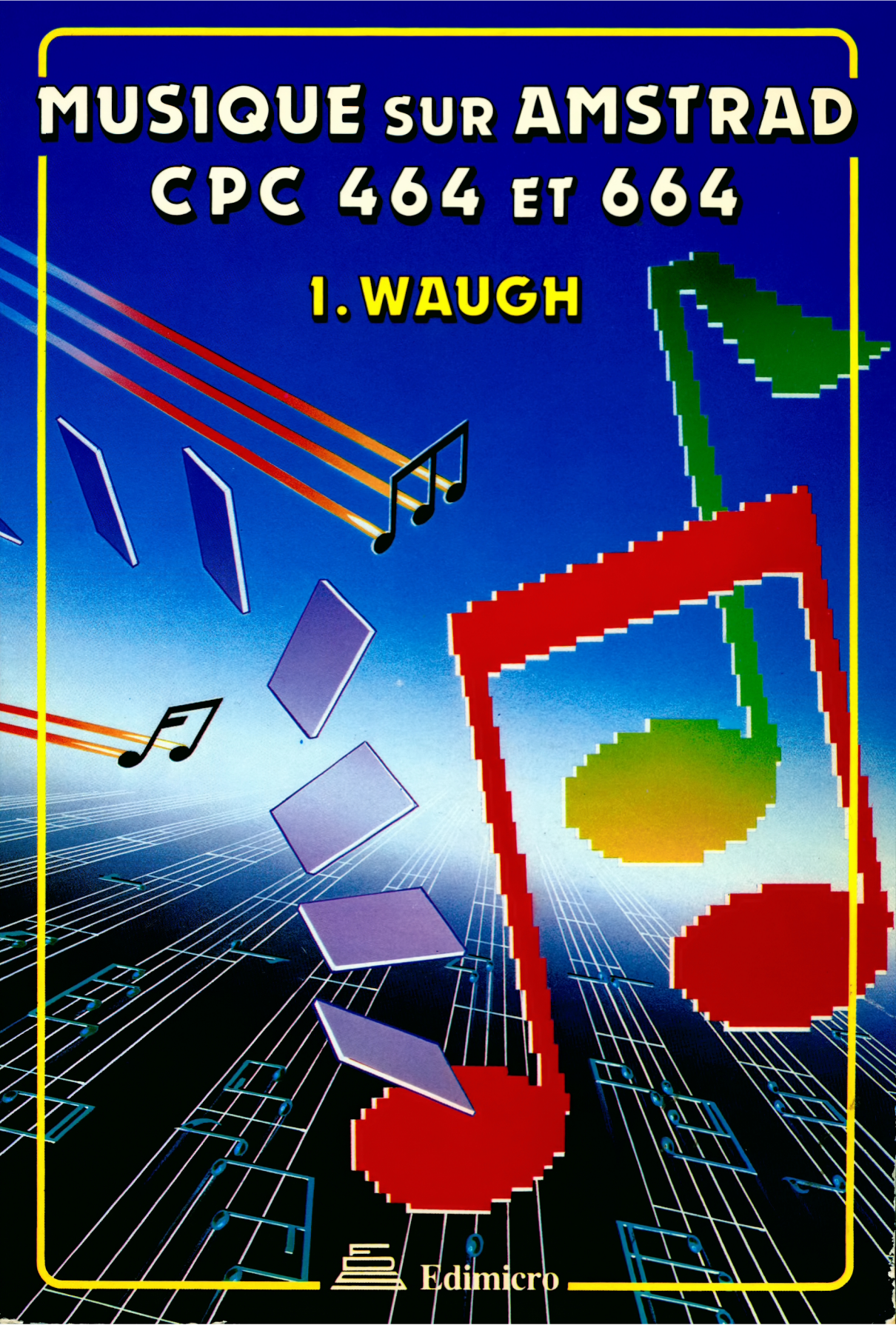
MUSIQUE SUR AMSTRAD CPC 464 ET 664

I. WAUGH



Edimicro

Musique sur Amstrad CPC 464 et CPC 664

Musique sur Amstrad CPC 464 et CPC 664

Ian WAUGH

Edimicro

Traduction autorisée de l'ouvrage publié en langue anglaise.
Original title : « Making Music on the Amstrad CPC 464 and 664 ».
Copyright © 1985 Ian Waugh.

First published in English 1985 by : Sunshine Books.
12-13, Little Newport Street, London WC2H 7PP.

Traduction et adaptation : Saisitex.

Couverture : Thierry Duval.

Imprimé en France. Droits mondiaux réservés.

« La loi du 11 mars 1957 n'autorisant, aux termes des alinéas 2 et 3 de l'article 41, d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective », et d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants droits ou ayants cause, est illicite » (alinéa 1er de l'article 40). » « Cette représentation ou reproduction, par quelque procédé que ce soit, constituent une contrefaçon sanctionnée par les articles 425 et suivants du Code pénal. »

© 1985 F.D.S./Edimicro pour la traduction française

ISBN 2-904457-41-0

EDIMICRO
DÉPARTEMENT ÉDITIONS DE F.D.S. SARL
121/127, avenue d'Italie, 75013 Paris

Sommaire

Introduction *1*

1 ***Qu'est-ce que le son ?*** *5*

LA NATURE D'UN SON 5
EXAMINONS LES ONDES SONORES 7
COMMENTAIRES 8
LE SON DE L'AMSTRAD 9
FRÉQUENCE : NOTES BASSES, NOTES HAUTES 10
LE VOLUME 12
LA DURÉE OU LONGUEUR DE NOTE 14
VITESSE ET PRÉCISION 15
TIMBRE OU QUALITÉ DU SON 17

2 ***Qu'est-ce que la musique ?*** *21*

LE LANGAGE MUSICAL 22
LA HAUTEUR D'UNE NOTE 23
LES GAMMES 26
LA GAMME MINEURE 29
LES NOTES ENHARMONIQUES 30
LES ACCIDENTS 30
LA DURÉE D'UNE NOTE 31
HARMONIES ET ACCORDS 35

3 ***La commande SOUND*** *39*

LE CANAL (C) 41
LE RENDEZVOUS 44
HOLD ET RELEASE 46
LA HAUTEUR (P) 48
LA DURÉE (D) 49
LE VOLUME (V) 50
L'ENVELOPPE DE VOLUME (VE) 50
L'ENVELOPPE DE TONALITÉ (TE) 52
LE BRUIT (N) 52
METTONS-LES TOUS ENSEMBLE 53
L'ÉTAT DE LA FILE D'ATTENTE SQ(c) 53
STÉRÉO ET HAUT-PARLEURS EXTÉRIEURS 56

4 ***Programmer notes et gammes*** *59*

CALCULONS LA HAUTEUR 59
LA GAMME TEMPÉRÉE 60
LA GAMME DE « JUSTE INTONATION » 61
LA GAMME DÉSACCORDÉE 62
LA GAMME MICROTONALE 65
CALCUL DES NOTES DANS LA GAMME TEMPÉRÉE 65
NUMÉROTATION DES NOTES ET OCTAVES 66
CONSTRUCTION MICROTONALE DES GAMMES 68
L'AVENIR DE LA MUSIQUE 69

5 ***ENV et ENT : les enveloppes de volume et de tonalité*** *71*

ENV : L'ENVELOPPE DE VOLUME 71
ADSR : ATTAQUE, DÉCLIN, SOUTIEN, RELÂCHEMENT 72
ENT : L'ENVELOPPE DE TONALITÉ 96
ENVELOPPES MULTI-SECTIONS 100
ZONE-MÉMOIRE DE STOCKAGE DES ENVELOPPES 101
STOCKAGE DES ENVELOPPES MATÉRIELLES 104
AUGMENTER LE NOMBRE DE SECTIONS DE L'ENVELOPPE 105
EXPÉRIENCES AVEC LES PROGRAMMES 107
CARACTÉRISTIQUES DES INSTRUMENTS 107

6 ***Variations musicales***

VIBRATO ET TREMOLO 109
PRODUCTION D'ÉCHOS 115
DIVERS EFFETS SPATIAUX 117
TINTEMENTS ET SONNERIES 120

7 ***Bruitages, sifflements et autres bizarreries*** *125*

LE BRUIT BLANC 125
EFFETS SONORES SIMPLES 126
ÉTUDE D'UNE UNITÉ RYTHMIQUE 128

8 ***Jouer sur l'Amstrad*** *137*

L'AMSTRAD, UN INSTRUMENT DE MUSIQUE 137
L'AMSTRAD, UN INSTRUMENT POLYPHONIQUE 138
L'AMSTRAD EN TANT QUE SYNTHÉTISEUR 140
UN SÉQUENCEUR DE BASSES 145
POUR CHANGER L'ACCOMPAGNEMENT 146
COMMENT AMÉLIORER LE SÉQUENCEUR 147

9 ***Composer de la musique sur micro-ordinateur*** *149*

PARLONS MUSIQUE 150
REMARQUE SUR LES NOTATIONS ANGLAISE ET FRANÇAISE 151
LA MÉTHODE NUMÉRIQUE 152
LA DURÉE DES NOTES 154
UN PROGRAMME DE CONVERSION NOTE/NOMBRE 155
LES ROUTINES D'ERREURS 158
ÉMISSION DE MÉLODIES COMPOSÉES 158
LA STÉRÉO 159
SUGGESTIONS ET MODIFICATIONS 166
DÉVERMINAGE DES DONNÉES 167
SAUVEGARDE DES INFORMATIONS 168
EXPÉRIENCES AVEC LES PROGRAMMES 170

10 ***La composition sur ordinateur*** *179*

LA COMPOSITION HUMAINE : ALGORITHMES ET HEURISTIQUE 180
L'ANALYSE DE LA NOTE EN COMPOSITION 189
PROGRAMME POUR ANALYSER TOUT LE MORCEAU 201

11 ***Votre composition sur Amstrad*** *203*

CHANSONS POPULAIRES : STRUCTURE HARMONIQUE 204
PRODUIRE DES RÉSULTATS ACCEPTABLES 205
ACCORDS ALÉATOIRES 206
UTILISEZ LES ACCORDS COMME BASE DE COMPOSITION 207
AJOUTEZ DES VARIATIONS DE RYTHME 216
AUTRES CONTRÔLES 217
EXTENSIONS ET MODIFICATIONS 218
AMÉLIOREZ LA MÉLODIE 220
CRÉEZ ET DÉVELOPPEZ VOS PROPRES PROGRAMMES 221
CHAHUTEZ VOS COMPOSITIONS 223
LA PROGRESSION VERLAN (À L'ENVERS) 226
LA CHANSON AUTOMATIQUE 228
MUSIQUE SUR L'AMSTRAD, DEMAIN 228

Annexe I *LES NOTES, LEUR FRÉQUENCE ET LEUR HAUTEUR*
Annexe II *ENTRÉE, PROTECTION ET EXÉCUTION DES PROGRAMMES*

Avertissement sur la notation musicale

Ce livre est traduit de l'anglais : il est donc utile, en préambule, d'évoquer les problèmes de conversion entre la notation anglaise et la notation française. Rassurez-vous, cela n'est pas bien compliqué : au lieu d'appeler les notes DO, RÉ, MI..., en anglais on se contente de leur donner une lettre de l'alphabet :

DO RÉ MI FA SOL LA SI ⟶ notation française
C D E F G A B ⟶ notation anglaise

Ainsi un F # 3 correspond à un FA # de la troisième octave. Pour vous familiariser avec les deux notations, nous introduirons la notation anglaise au fur et à mesure. En effet, les programmes permettant de faire la correspondance entre une notation conventionnelle et un nombre ont tous été réalisés à partir de la notation anglaise. Pour adapter les programmes à la représentation française, il aurait fallu retranscrire tous les programmes. A notre avis, le marché de l'informatique étant pratiquement entre les mains des Anglo-Saxons, il est nécessaire de savoir transcrire un SOL en G, et un A en LA. Au début nous donnerons les deux représentations pour faciliter la transcription.

Introduction

Ce livre est spécialement destiné aux possesseurs d'un ordinateur AMSTRAD. Les motivations qui vous ont poussé à l'achat d'un ordinateur sont sûrement très diverses, mais en arrêtant votre choix sur un AMSTRAD vous êtes entré en possession d'un ordinateur puissant et doué d'une grande souplesse d'utilisation.

Plusieurs rayonnages de livres seraient nécessaires si l'on voulait examiner en détail toutes ses particularités. Ce manuel analyse l'une de ses plus passionnantes caractéristiques: LE GENERATEUR DE SON ou SYNTHETISEUR.

Le fait d'intégrer un circuit permettant la génération des sons dans un ordinateur personnel est assez récent. Il contribuera certainement à l'intérêt croissant que porte le public aux micro-ordinateurs.

Plus de 75% des informations que nous percevons du milieu extérieur viennent du sens de la vue, et il ne faut donc guère être surpris si toute la publicité des micro-ordinateurs est axée sur leurs possibilités graphiques.Mais quel peut être l'impact d'une explosion de missile sans le bruit de la détonation, ou d'une course de hors-bords sans la plainte stridente des moteurs? Nous avons souvent du mal à mesurer les effets de la musique ou des bruits sur notre comportement. Faites l'expérience de couper le son de votre téléviseur lors du prochain film d'épouvante ou d'une poursuite de voitures et

vous pourrez apprécier à quell degré l'environnement sonore participe à l'émotion.

La génération des bruits n'est pas la seule particularité de ce micro; il peut également jouer de la musique. Ceci nous ouvre un domaine totalement nouveau dans le monde de la micro.

L'AMSTRAD a la possibilité de produire une séquence ordonnée de notes ,des sons harmonieux avec en plus toutes les caractéristiques d'un ordinateur: vitesse, analyse logique et décision.

A propos de ce livre

Le but de ce livre est d'agir comme un tremplin dans le développement et l'écriture de vos prochains programmes. L'accent est mis sur les bruitages, les sons, la musique, et la manière de tirer le meilleur profit du générateur de son de l' AMSTRAD.

Les programmes sont facilement compréhensibles, ils sont structurés et documentés. Simples, sans commentaires inutiles, sans fioritures, ils vous permettent de réduire au minimum le temps nécessaire à la frappe, et d'éviter les erreurs. L'aspect global d'un programme est aussi important que ses performances, et de nombreuses suggestions sur ce point seront avancées tout au long du livre. D'autre part, vous pourrez toujours par la suite faire des modifications pour apporter votre contribution ou votre "touche personnelle". Vous trouverez également quelques suggestions pour les futurs développements de programmes, mais le sujet est tellement vaste et complexe que pour traiter à fond tous les domaines il aurait fallu écrire un second livre.

Ce livre a été conçu de manière progressive, mais il n'est pas indispensable de commencer au chapitre 1; vous pouvez immédiatement vous plonger dans le chapitre qui vous intéresse.

Pour faciliter la compréhension des novices, de nombreuses répétitions apparaissent dans les différents chapitres, ainsi que des renvois. Que les spécialistes en musique ou en programmmation nous le pardonnent; les profanes nous en sauront gré.

Que vous cherchiez un nouveau son laser pour votre dernier jeu ou une tonalité pour votre sonnette, que vous utilisiez l'ordinateur pour apprendre le solfège ou tout simplement pour voir ce que l'on peut tirer d'un générateur de son, ce livre présentera quelque intérêt pour vous.

Avant de commencer,nous vous invitons à vous reporter à l'annexe 2 qui contient, entre autres, des conseils et des astuces pour entrer et mettre bout à bout vos programmes.

Remerciements de l'auteur

Je voudrais remercier une fois de plus David Lawrence pour ses conseils et ses suggestions.

Merci également à Bill Poel, d'AMSOFT, pour sa coopération, et à la société AMSTRAD pour avoir conçu un micro-ordinateur aussi fabuleux.

Je ne voudrais pas oublier ma mère et mon père qui m'ont soutenu et encouragé dans toutes mes entreprises.

1

Qu'est-ce que le son ?

Il va de soi que dans ce chapitre nous n'étudierons le son que dans le domaine de la micro-informatique.

Le BASIC Amstrad est un langage puissant et rapide. Il contient de nombreuses commandes permettant de programmer le générateur de son. Cependant, ces commandes sont complexes et d'une utilisation plutôt difficile. La complication provient surtout des paramètres attachés à chaque commande et aux valeurs par défaut que peuvent prendre ces derniers. Nous étudierons la manière de programmer sans peine ce générateur en utilisant uniquement les paramètres nécessaires.

Le guide d'utilisation ne consacrant que peu de pages à la génération des sons, d'autres informations sont nécessaires pour en tirer le meilleur profit. Le problème est exactement le même quand on aborde la programmation BASIC pour la première fois: l'ordinateur a son propre langage, et nous devons absolument nous adapter à ce langage pour contrôler la machine; or, avec le générateur de son, il nous faut également traduire les caractéristiques d'un son dans un langage que l'ordinateur peut comprendre.

La nature d'un son

Nous sommes souvent en difficulté pour comprendre un son car nous sommes confrontés à

un domaine invisible. Un son est émis par friction ou percussion d'un objet, en termes scientifiques on dirait: "mis en vibration". Ces vibrations sont transmises à l'air environnant qui les transporte jusqu'à notre oreille. L'air n'est pas le seul véhicule des vibrations; les gaz, les liquides, les solides peuvent jouer le même rôle. En frappant la tôle d'un bidon plein d'eau vous pouvez observer les vibrations générées.

Dans l'espace, où règne le vide complet, il n'existe aucun support pour transmettre les vibrations, et le silence est complet. Les sons laser émis par les véhicules spatiaux dans les films de fiction n'ont aucune réalité scientifique, et les attaques de ces engins devraient se faire dans le plus grand silence. Mais ne supprimez pas tous les sons de vos jeux interstellaires; tant de réalisme serait ennuyeux.

Les instruments de musique produisant une tonalité par percussion sont: la batterie, le piano, le gong et le xylophone. Le violon, lui, produit un son par friction d'une corde. Dans les cuivres (trompettes, trombones), le son est produit par le souffle et la vibration des lèvres. L'air contenu dans l'instrument est mis en vibration à une fréquence proportionnelle à la longueur du tube de cuivre. La flûte est basée sur un principe identique, la vibration de la colonne d'air contenue dans le tube se faisant au travers du bec. On peut retrouver un effet similaire en soufflant dans le goulot d'une bouteille. Dans les clarinettes et saxophones la vibration des lèvres est amplifiée par la vibration d'une anche.

Dans notre micro-ordinateur le son est généré à partir d'un signal électrique de fréquence variable. Plus la tonalité à reproduire est haute, plus la fréquence est élevée. Ces oscillations sont transmises à un haut-parleur qui met en vibration l'air environnant.

Examinons les ondes sonores

Les vibrations sonores se déplacent en une série d'ondes, et chaque son produit sa propre forme d'onde. Si nous branchons un oscilloscope à la sortie d'un microphone, nous pouvons très bien matérialiser la forme des ondes.

Un signal sinusoïdal délivre un son pur, sur une seule tonalité ou fréquence. Un tel signal est généralement émis par un diapason ou un générateur électronique. Il est possible de produire de nombreux sons différents en combinant plusieurs signaux sinusoïdaux. Ce procédé est connu sous le nom de synthèse additive puisque les fréquences sont ajoutées les unes aux autres dans des proportions bien connues. Quelques synthétiseurs du commerce utilisent ce procédé, malgré son coût élevé et sa lenteur de mise en oeuvre.

Le programme suivant va tracer une onde sinusoïdale; l'amplitude dépendra du volume de votre son, et la fréquence du signal de votre paramètre d'entrée. Il permet de visualiser comment la fréquence et l'amplitude affectent un signal électrique, et nous l'utiliserons un peu plus loin dans ce chapitre pour tracer le signal résultant d'une synthèse additive.

```
100 REM PROGRAM 1.1
110 REM Traceur d'onde sinusoïdale
120 :
130 MODE 1
140 WINDOW #0,1,40,1,5
150 ORIGIN 0,160,0,639,319,0
160 REM Fond d'écran noir, graphique en
    blanc
170 INK 2,0:INK 3,13:CLG 2
180 DEG
190 :
200 WHILE k$<>"F"
210 INPUT "Fréquence (1-20)";freq
220 INPUT "Amplitude (5-150)";amp
230 GOSUB 290:REM Tracer une onde
```

```
240 PRINT"Presser une touche quelconque
    pour tracer une autre onde" "'C'
    pour effacer l'écran, 'F' pour terminer"
250 k$="":WHILE k$="":k$=UPPER$(INKEY$):
    WEND
260 IF k$="C" THEN CLG 2
270 WEND
280 END
290 :
300 REM Tracer une onde
310 MOVE 0,0
320 FOR stime = 1 to 640
330 DRAW stime,amp*SIN(freq*stime),3
340 NEXT stime
350 RETURN
```

Commentaires

Les lignes 320 à 340 contiennent le programme de tracé; la ligne 330, que nous améliorerons plus tard, exécute les calculs.

Tout d'abord, essayez d'entrer 1 pour la fréquence et 100 pour l'amplitude. Si vous augmentez la fréquence, vous pouvez vous apercevoir que les ondes sont de plus en plus serrées. L'unité de fréquence est le nombre de cycles par seconde, ou hertz (Hz); plus la fréquence est élevée, plus il y a de cycles par seconde ou hertz. Une augmentation d'amplitude va agrandir notre signal électrique sans en affecter la fréquence, en d'autres termes le volume augmente mais la tonalité reste la même.

Si vous remplacez la ligne 330 par:

```
330 DRAW stime,amp*SIN(freq*stime)+amp*
    SIN(freq*2*stime),3
```

vous allez faire l'addition de deux fréquences et vous pourrez voir le résultat d'une synthèse additive. Notez bien le changement de la forme du signal.

On peut ajouter plusieurs fréquences en ligne 330, en modifiant la variable **freq** dans

l'expression:

```
amp*SIN(freq*stime)
```

Si vous modifiez la valeur **amp** de la manière suivante:

```
amp*.5*SIN(freq*stime)
```

vous allez réduire l'amplitude de la première onde ainsi que son impact sur la tracé résultant qui va être diminué. Ces additions de fréquences sont connues sous le nom d'harmoniques et c'est elles qui différencient les sons. La plupart des sons que nous percevons dans notre vie quotidienne ont des formes d'onde très complexes et sont composés d'une multitude de fréquences d'amplitudes diverses. D'autres exemples seront donnés dans ce chapitre quand nous aborderons le timbre.

Le son de l'Amstrad

Il existe une autre forme de synthèse appelée synthèse soustractive et qui consiste à prendre un son complexe et à en filtrer certaines harmoniques. Cette méthode est la plus utilisée dans les synthétiseurs. Le contrôle de tonalité est un simple filtre qui retire les fréquences les plus élevées.

Mais avant de filtrer un signal, il faut déjà avoir quelque chose à filtrer. Une sinude, constituée d'une seule fréquence, ne nous sera pas d'une grande utilité. Les meilleurs signaux sont ceux contenant un grand nombre d'harmoniques car ils nous permettent de trier correctement. Généralement, les synthétiseurs travaillent à partir de formes d'ondes différentes telles que des ondes carrées, triangulaires ou en dents de scie. Une onde triangulaire est très proche d'une onde sinusoïdale, à cette différence près qu'elle contient de nombreuses harmoniques. Les autres formes d'ondes produisent des sons un peu différents: l'onde carrée

se rapprocherait du son de la clarinette, l'onde en dents de scie de celui du pipeau.

Essayez l'exemple suivant:

```
10 FOR hauteur=1600 to 20 STEP -10
20 SOUND 1,hauteur
30 NEXT hauteur
```

Votre AMSTRAD joue quelques éléments de son répertoire - d'une manière un peu discontinue, soit! Mais cet exemple nous sert de support pour montrer le registre des fréquences que peut produire un générateur de son. Il n'est pas évident de déterminer la forme des ondes utilisées; dans les fréquences basses, le son émis est assez proche de celui d'une râpe, et dans les fréquences élevées, il ressemblerait plutôt à un tintement métallique. Les fréquences moyennes, par contre, sont harmonieuses. Si vous changez la ligne 10:

```
10 FOR hauteur=500 to 100 STEP -10
```

le son généré ressemble à celui d'une clarinette; vous êtes sur le bon chemin car le signal de base utilisé par le micro est un signal carré.

Pour reproduire correctement un son il faut jouer sur quatre paramètres:

1) la tonalité
2) le volume
3) la durée
4) le timbre.

Nous allons examiner ces quatre aspects du son et déterminer quel est leur rapport avec les instruments de musique.

Fréquence : Notes basses, notes hautes

La tonalité d'une note est déterminée par sa position haute ou basse dans le spectre musical, et le terme de fréquence y est souvent associé.

A proprement parler, ce dernier se rattache plutôt à la forme d'onde pour nous indiquer à quel nombre de cycles par seconde ou hertz elle oscille. L'oreille humaine est capable d'entendre des sons de 20 à 20 000 Hz, la limite supérieure décroissant rapidement avec l'âge.

Une mélodie est constituée d'une série de sons ayant des relations bien précises. Dans la musique occidentale, la mélodie est basée sur des tons que nous pouvons jouer au piano, les notes étant séparées d'un demi-ton entre elles. Les notes sont regroupées dans des gammes: DO majeur, SI mineur.

Au piano, la tonalité des notes est fixée et il est difficile de faire des couacs sans désaccorder votre piano. D'autres instruments tels que les cuivres demandent un contrôle des tonalités, et certaines notes peuvent être chaînées. Quand cet artifice est répété et voulu, on l'appelle un portamento et ce genre de son peut être facilement généré sur un synthétiseur. Sur un piano, une harpe ou un xylophone le même effet peut être obtenu en balayant toutes les cordes ou touches, donc par une suite de demi-tons discrets appelé glissando. Ces effets sont surtout utilisés par les musiciens de jazz et peuvent apporter une touche personnelle à une musique synthétique. Le programme suivant va nous montrer cet effet:

```
100 REM PROGRAM 1.2
110 REM Démonstration de Portamento &
    Glissando
120 :
130 FOR hauteur=478 TO 239 STEP -1
140 SOUND 1,hauteur,1
150 NEXT hauteur
160 :
170 FOR d=1 TO 1000:NEXT d
180 :
190 FOR note=0 TO 12
200 freq=261.626*(2^(note/12))
210 hauteur=ROUND(125000/freq)
220 SOUND 1,hauteur,10
230 NEXT note
```

Le portamento est obtenu en passant très rapidement les fréquences de 470 à 239. La note est maintenue 1/100 de seconde et programmée par le dernier paramètre de la commande SOUND en ligne 140. Si vous ôtez ce paramètre de la commande, vous passez à la valeur par défaut qui est de 1/5 de seconde ou 200 millisecondes et vous pourrez entendre les notes monter plus lentement.

L'effet portamento est obtenu par un défilement de fréquences si rapide que votre oreille les amalgame toutes ensemble. L'exemple du glissando produit une série de demi-tons autour d'une même fréquence. Les lignes 200 et 210 calculent les fréquences. Une description plus détaillée des paramètres de la commande SOUND sera faite au chapitre 3 et la formule utilisée pour les fréquences sera revue au chapitre 4. Vous pouvez aller jeter un coup d'oeil pour voir la manière de programmer, les exemples sont vraiment très faciles à suivre.

L'AMSTRAD peut produire une grande variété de mélodies et de sons du type occidental ou autre. Dans la musique orientale les notes sont séparées par moins d'un demi-ton, c'est pourquoi la mélodie est très différente.

Le volume

C'est avec ce terme que l'on définit la puissance sonore d'un son. A première vue, ce paramètre semble être moins important que les autres et surtout plus facile à manier. Mais plusieurs facteurs affectent la perception du volume sonore.

La réverbération, l'écho, le vibrato et la durée sont des éléments qui tendent à augmenter le volume de la même manière que l'addition des harmoniques. Par exemple, un son maintenu pendant 1/100 ou 1/10 de seconde ne semble pas aussi fort qu'un son maintenu pendant 1 seconde.

Dans nos exemples nous avons également pu remarquer que le volume est affecté par la fréquence. Ceci résulte des propriétés des sons et du synthétiseur et vous ne pouvez corriger ce défaut qu'en jouant sur les instructions pour tenir compte des variations de volume, mais cela ne pose pas un grand problème.

L'amplitude d'un son varie dans le temps. Le piano,le xylophone ou les autres instruments de percussion produisent une impulsion sonore qui atteint son maximum presque immédiatement, puis meurt peu à peu. Une attaque au violon est un peu moins incisive et il faut un temps plus long pour atteindre l'amplitude maximale. Les cuivres atteignent toute leur plénitude sonore très rapidement même en jouant des sons très faibles, les poumons étant gonflés à l'avance. Cette variation de volume est appelée l'enveloppe sonore et joue un rôle déterminant dans les caractéristiques instrumentales. Essayez le programme suivant:

```
100 REM PROGRAM 1.3
110 REM Démonstration de volume
120 :
130 FOR vol=1 TO 7
140 SOUND 1,478,5,vol,1
150 NEXT vol
160 :
170 FOR d=1 TO 1000:NEXT d
180 :
190 FOR vol=7 TO 1 STEP -1
200 SOUND 1,478,5,vol
210 NEXT vol
```

Le premier exemple produit un son assez bizarre ressemblant à celui d'une bande magnétique passant à l'envers. Ceci est dû à la montée en crescendo et à l'arrêt brutal du générateur de son. Dans le second exemple nous retrouvons la sonorité d'un instrument de percussion frappé doucement. Si les deux séquences sont mises bout à bout en ôtant la ligne 170, nous obtenons quelque chose de plus harmonieux.

Le fait de produire des sons "inversés", c'est-à-dire avec un arrêt brutal au maximum du volume, est très utilisé dans les synthétiseurs et nous pouvons l'exploiter sur l'AMSTRAD pour produire des effets intéressants.

Plutôt que de contrôler le volume avec une boucle FOR/NEXT nous pouvons utiliser la commande ENV pour créer un jeu prédéterminé de volumes.

```
100 REM PROGRAM 1.4
110 REM Démonstration d'ENV
120 :
130 ENV 1,16,15,10
140 :
150 FOR note=0 TO 12
160 freq=261.626*(2^(note/12))
170 hauteur=ROUND(125000/freq)
180 SOUND 1,hauteur,100,0,1
190 NEXT note
```

Ceci génère une enveloppe percussive et produit une sonorité de piano.

En changeant la ligne 130:

```
130 ENV 1,16,17,4
```

nous obtenons notre son de tout à l'heure, et s'il ne s'arrêtait pas d'une manière aussi soudaine il pourrait former une enveloppe sonore du type violon. ENV est expliqué au chapitre 5 et l'un des programmes contenus dans ce chapitre vous montrera graphiquement la manière dont les commandes ENV et ENT affectent l'amplitude et la fréquence.

La durée ou longueur de note

Nous abordons à nouveau une notion complexe. Pour qu'un son puisse exister il lui faut une certaine durée. En ce qui concerne l'AMSTRAD, la durée d'un son ne pourra être inférieure à 1/100 de seconde. Ceci est le temps minimum

que vous pouvez assigner à une note dans la commande SOUND, les paramètres de temps dans les commandes ENV et ENT étant exprimés en millisecondes. Ceci nous permettra de contrôler complètement les sons produits, et aussi de nous adapter à toutes les mélodies. D'un point de vue psychologique, il est intéressant de noter la différence de perception du temps d'un individu à l'autre. Suivant l'environnement, le temps semble s'écouler plus ou moins rapidement. De nombreux conférenciers ennuyeux ont l'impression de tenir des discours d'un quart d'heure, alors qu'ils tiennent les gens en haleine pendant plus d'une demi-heure. Si vous interrogez l'auditoire, il aura l'impression d'avoir passé trois quarts d'heure à l'écouter. Tout ceci pour montrer que l'appréciation de la mesure n'est pas évidente et que contrairement à l'appréciation musicale elle n'est normalement pas dépendante de l'oreille.

Le sens du rythme joue une grande part dans la création de partitions musicales. Les seuls éléments que le pianiste doit contrôler sont le volume et la mesure. Le timbre et les notes sont déterminés par l'instrument et le compositeur; les grands pianistes sont tout de même jugés sur leur sens du rythme, sur l'accentuation et le phrasé de leur partition.

Vitesse et précision

La coordination est souvent citée comme un trait de grande importance pour un musicien. La base de toute prestation musicale est la précision et la vitesse d'exécution. Nous trouvons des personnes rapides et précises, ou précises mais lentes, ou non précises, ou encore manquant des deux qualités. Il existe une limite naturelle à la vitesse d'exécution, mais cela n'est pas déterminant pour juger de la classe d'un musicien; par contre, les changements de mesure permettent déjà de mieux apprécier.

Sans trop perdre de vue l'objet de ce livre nous pouvons tout de même programmer un petit test destiné aux amateurs de musique et déterminant leur vitesse et leur précision. Il servira aussi à tous ceux qui veulent développer leurs réflexes car cela pourra augmenter leurs performances dans les jeux de poursuite. Votre rapidité de réaction peut être mesurée en tapant sur une touche et en comptant le nombre de frappes par seconde. Le programme suivant comptabilise le nombre de touches frappées dans un laps de 5 secondes.

```
100 REM PROGRAM 1.5
110 REM Test de rapidité
120 :
130 KEY DEF 18,0:REM Départ répétition
    touche RETOUR
140 ENT -1,=478,10,=379,10,=358,10,=319,
    10
150 score=0
160 CLS
170 LOCATE 5,10:PRINT"Frapper la touche
    RETOUR plusieurs fois","  aussi
    rapidement que possible et avec","
          le mouvement minimal."
180 WHILE k$="":k$=INKEY$:WEND
190 tim=TIME+1500
200 :
210 WHILE TIME<tim
220 k$=INKEY$:IF k$=CHR$(13) THEN score=
    score+1
230 WEND
240 :
250 SOUND 1,0,90,7,0,1
260 LOCATE 18,14:PRINT"STOP":PRINT
270 PRINT TAB(9)"Votre score de rapidité
    est de"
280 PRINT TAB(10) score/5;"coups par
    seconde."
290 PRINT:PRINT TAB(11)"Encore une fois
    (O/N)?"
300 k$=UPPER$(INKEY$):IF k$="O" THEN RUN
310 IF k$<>"N" GOTO 300
320 END
```

Commentaires

La partie la plus importante du programme se situe entre les lignes 210 et 230 qui incrémentent la variable score quand la personne frappe la touche ENTER. Cette touche a été choisie car elle n'est pas répétitive; malgré tout, la ligne 130 vérifie. Remplacez cette ligne par:

```
130 KEY DEF 18,1
```

et vous verrez la différence.

Les instructions ENT et SOUND, en lignes 140 et 250, produisent une petite musique sans utiliser un grand nombre de données. Les curieux et les impatients trouveront au chapitre 5 des précisions sur cette manière de programmer. Il est évident que la pratique de ce test va accroître votre score. 8,5 coups par seconde est une bonne moyenne au début, et après une ou deux semaine d'expérience vous devriez atteindre 9,3. Il semblerait que les hommes soient un tout petit peu plus rapides, d'un demi-point, que les femmes. Pour qu'un tel exercice puisse tester valablement les amateurs de musique il faut recréer leur environnement: le test du pianiste doit se faire sur une touche de piano et celui du violoniste sur une corde. Les meilleures performances représentent 15 coups par seconde.

Le timbre ou qualité du son

Le timbre est la coloration d'un son, il permet de distinguer une même note jouée par deux instruments différents. Le timbre est dépendant de la note et de l'enveloppe sonore, nous savons par exemple que les notes les plus basses de la clarinette ont une sonorité différente de celle des notes les plus hautes.Le synthétiseur de l'AMSTRAD réagit exactement de la même manière, ainsi que nous avons pu nous en apercevoir.

La coloration sonore est la combinaison des harmoniques dans un son. Nous avons vu les effets dans le premier programme "Sine Wave Program". Rechargeons le programme et passons en REM la ligne 330. Nous allons ajouter la ligne 334 du programme suivant et faire RUN. Par la suite nous passons en REM cette ligne et ajoutons la ligne 336.

```
100 REM PROGRAM 1.6
110 REM Synthèse de sons additionnels
120 :
333 REM Dents de scie
334 DRAW stime,amp*SIN(freq*stime)+amp*1
    /2*SIN(freq*2*stime)+amp*1/3*SIN
    (freq*3*stime)+amp*1/4*SIN(freq*4*
    stime)+amp*1/5*SIN(freq*5*stime),3
335 REM Ondes rectangulaires
336 DRAW stime, amp*SIN(freq*stime)+amp*
    1/3*SIN(freq*3*stime)+amp*1/5*SIN
    (freq*5*stime)+amp*1/7*SIN(freq*7*
    stime)+amp*1/9*SIN(freq*9*stime),3
```

Le premier exemple, en ligne 334, ajoute les harmoniques 2, 3, 4 et 5 à la fréquence fondamentale. Celle-ci est naturellement la plus grande, donc de la plus forte sonorité. Le fait d'ajouter toutes ces harmoniques transforme le signal en dents de scie. Plus on ajoute d'harmoniques, plus la sinusoïde se transforme en dents de scie.

La ligne 336 ajoute les harmoniques impairs, dans ce cas on obtient un signal carré. Plus nous ajoutons d'harmoniques, mieux le signal sera dessiné.

Nous pourrions multiplier les expériences en mixant les harmoniques ou en partant d'une courbe cosinusoïdale pour produire des ondes complexes. Il existe des livres spécialisés dans la synthèse des sons qui vous donneront tous les détails pour recréer n'importe quelle forme d'onde.

Sur l'AMSTRAD nous n'avons pas le contrôle de la forme d'onde, mais en utilisant avec un peu d'intelligence les commandes ENT, ENV et SOUND nous pouvons ruser et faire sortir des sons qui apparaîtront différents de ceux que génère une onde carrée. L'oreille est aussi sensible à l'enveloppe sonore qu'au timbre. Il y a cependant des limites dans ces petites astuces et nous allons explorer tout cela au long de ce livre.

2

Qu'est-ce que la musique ?

On a souvent dit que la musique est un compromis entre le chaos et la monotonie, et il ne faut pas chercher bien loin des exemples répondant à ces deux extrêmes.

De nombreux musiciens sont familiarisés avec les ordinateurs, mais pas nécessairement avec ceux comportant un synthétiseur. D'autre part tous les utilisateurs d'un micro-ordinateur ne sont pas amateurs de musique et jugent trop souvent que le temps nécessaire à apprendre le solfège est trop long, pour uniquement faire joujou avec le générateur de son...

Tout au long de ce livre nous parlerons musique et comme cet ouvrage n'est pas uniquement réservé aux musiciens je me dois d'expliquer quelques rudiments de musique. De nombreux livres ont été écrits sur le sujet et il serait insensé de résumer ces ouvrages en quelques lignes. Cependant, il est intéressant de présenter quelques règles et quelques références dans ce livre, pour les non-initiés.

Si vous lisez ce livre vous avez probablement un intérêt quelconque pour la musique. Ce chapitre a pour but de vous donner des informations suffisantes pour arriver à lire une partition, la transcrire en programme AMSTRAD et comprendre ce que vous faites.

Le langage musical

S'initier à la musique revient au même que d'apprendre une langue étrangère mais demande un peu moins d'efforts. Si votre ambition est de devenir un grand pianiste et qu'à ce jour le solfège vous est complètement inconnu, il est probablement trop tard, mais si vos ambitions sont limitées il est encore temps,et la connaissance de la musique vous apportera de nombreuses satisfactions.

L'un des problèmes, pour les novices, est la vue de ces pages noires de lignes et de points qui se ressemblent tous. Si votre objectif est de savoir interpréter une partition sur un instrument il vous faut absolument étudier le solfège, par contre pour les propos de ce livre vous n'avez besoin que de savoir comprendre les idées et les principes cachés par ces points noirs. Si vous voulez poursuivre l'étude du solfège il existe des dizaines d'excellents ouvrages.

Nous savons tous à quoi ressemble la musique, même si nous ne pouvons l'interpréter, et vous devez penser qu'il existe bien des moyens plus simples pour représenter les notes.

Dans nos programmes nous ne ferons pas tout le temps référence à la notation conventionnelle, mais il est utile de savoir ce qu'elle est et ce qu'elle représente. Vous trouverez les idées musicales plus faciles à suivre et cela augmentera votre appréciation des différents effets musicaux. Vous serez également en mesure de les traduire sous forme de programme.

La musique est écrite de cette manière pour répondre à une certaine convention. C'est simplement la voie utilisée depuis des siècles et l'avantage de cette représentation (les mauvaises langues disent: "le seul avantage") est d'être reconnue universellement. Si votre intention est de ne pas communiquer, personne ne peut vous empêcher de développer votre propre

notation musicale. En effet, il existe plusieurs autres formes de transcription musicale, toutes supposées plus faciles à apprendre, mais à ce jour aucune n'a pu atteindre une notoriété suffisante et elles sont donc toutes limitées dans le domaine de la communication, qui est essentiel en musique. En attendant que d'autres formes de notation se répandent, nous allons concentrer nos efforts sur la notation conventionnelle.

Les deux informations les plus importantes que nous obtenons d'une partition sont la hauteur d'une note et la durée pendant laquelle elle doit être jouée. Jetons un regard à la représentation de la note.

La hauteur d'une note

Dans la notation conventionnelle, les notes sont placées sur un ensemble de cinq lignes appelé portée. La hauteur d'une note est indiquée par son placement sur les lignes ou dans les interlignes. Plus la note est placée haut plus le son qu'elle représente est aigu. Pour différencier les notes on les affecte d'une lettre, comme dans les pays anglo-saxons: A B C, etc., ou d'un nom, comme en France: LA,SI,DO, etc. Quand vous atteignez B ou SI, vous recommencez à la première note (C ou DO). Pour accroître les possibilités, les notes peuvent être placées sur des lignes supplémentaires, en dessous ou au-dessus de la portée. Ces notes sont placées sur des tout petits traits qui sont en fait une extension de la portée. La portée pourrait être constituée de dix lignes ou plus, mais la lecture en deviendrait difficile. Le nombre de petits traits peut être accru autant que vous le désirez, mais s'il y en trop nous tombons dans le même inconvénient.

Pour augmenter l'étendue des notes et en assurer une lecture facile nous pouvons ajouter une seconde portée à la première et la placer en

dessous. Pour distinguer une portée d'une autre on l'affecte d'une clef différente. Les clefs les plus utilisées, et les seules qui nous concernent, sont la clef de SOL et la clef de FA. La clef de SOL désigne la portée supérieure, la clef de FA la portée inférieure (voir figures 2.1 et 2.2).

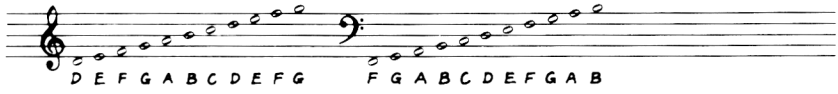



Figure 2.1. Notes sur la portée.

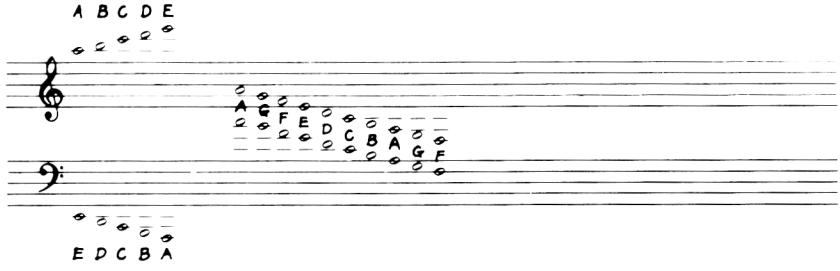



Figure 2.2. Lignes supplémentaires.

Sur ce dessin vous pouvez également remarquer que les notes de la clef de SOL sont descendantes et les notes de la clef de FA ascendantes, en fait elles correspondent. Une partition de piano est normalement écrite en clef de SOL et en clef de FA, mais il est possible de trouver deux clefs de SOL ou de FA, l'une en dessous de l'autre.

Si ces deux clefs ne sont pas suffisantes, on a recours à d'autres solutions, telles que le changement de clef ou l'OCTAVA. Ce dernier procédé consiste à écrire les notes une octave plus basse ou plus haute en les surmontant d'un crochet pointillé continu précédé d'un 8va sup ou 8va bassa (voir figure 2.3). Ceci devrait couvrir tous les besoins.

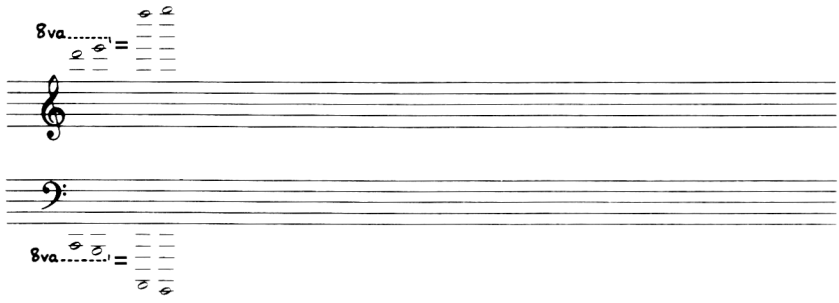


Figure 2.3. Exécution d'un octave plus haut ou plus bas.

L'intervalle séparant la hauteur de deux notes de même nom, deux RE par exemple, est appelé une octave; elle représente un doublement de la fréquence de la note. L'intervalle entre une note sur une ligne et une note entre les lignes peut être d'un ton ou d'un demi-ton. Dans la démonstration du glissando et du ENV (programmes 1.2 et 1.4) nous avions une suite de demi-tons. Même si vous n'êtes pas experts, vous pouvez deviner que la gamme n'est pas complète. Le programme suivant va jouer une gamme en DO en démarrant du DO en dessous du LA3 du diapason.

```
10 FOR gamme =1 TO 8
20 READ note
30 SOUND1,note
40 NEXT gamme
50 END
60 DATA 478,426,379,358,319,284,253,239
```

Ce programme nous joue une séquence musicale plus agréable à écouter qu'une suite de demi-tons. Les notes sont déterminées dans l'instruction DATA et la progression des intervalles est de: ton,ton,demi-ton,ton,ton,ton,demi-ton. Cette séquence est la même pour toutes les gammes majeures, vous obtenez le même résultat en jouant toutes les notes blanches de manière ascendante. Il existe d'autres gammes que les gammes majeures que nous verrons un peu plus loin.

Si tout ceci est nouveau pour vous, n'essayez pas de tout assimiler du premier coup; au contraire, après une première lecture essayez de revenir à ce chapitre dès que vous rencontrez un problème. La figure 2.4 devrait vous aider; cette figure nous montre un clavier de piano, les indications des notes et leur représentation sur la portée. Nous trouvons également la fréquence de chaque tonalité, information utile pour la commande SOUND. Les nombres utilisés par l'AMSTRAD ont une relation spéciale avec le son produit mais nous nous étendrons sur ce sujet dans le chapitre 3.

Le générateur de l'AMSTRAD a une étendue de huit octaves. La figure 2.4 ne montre que les cinq octaves centrales pour ne pas trop être encombrées et ces octaves, d'ailleurs, sont les plus utilisées. En cas de nécessité nous pouvons facilement trouver les informations nécessaires pour programmer les autres octaves.

Les gammes

C'est un peu la hantise de tous les débutants, car l'étude des gammes est souvent synonyme d'ennui. En étudiant l'AMSTRAD nous ne sommes pas obligés de passer par ce genre d'exercice quotidien pour notre pratique musicale. Nous devons seulement nous souvenir qu'il existe douze notes différentes dans le spectre musical, et si nous voulons atteindre la treizième note la séquence est répétée dans l'octave supérieure.

Le programme suivant nous permet de tester tout cela :

```
10 hauteur=3822
20 FOR oct=-3 to 4
30 PRINT "Octave =";oct
40 SOUND 1,hauteur,100
50 hauteur=hauteur/2
60 NEXT
```

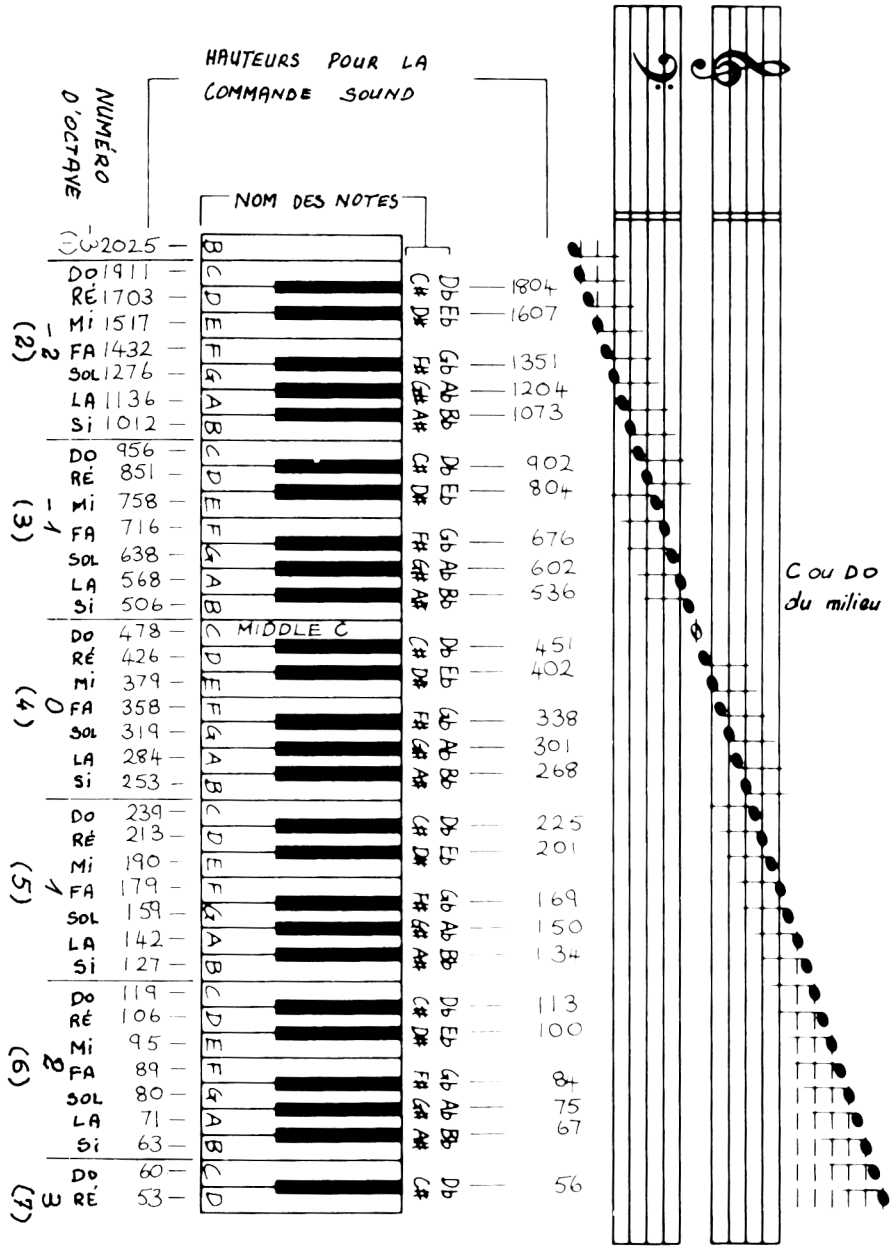


Figure 2.4. Le clavier, le nom des notes et leur période SID 16 bits.

Les notes au-dessous de l'octave -2 émettent des grondements et des roulements de tambour, celles de l'octave 3 sont très faibles. Nous disions précédemment que le fait de passer d'une octave à l'autre représente un doublement de la fréquence. La figure 2.4 nous montre deux jeux de numérotation des octaves, la supérieure est celle utilisée dans le Guide d'Utilisation de l'AMSTRAD.La ligne inférieure y est ajoutée pour des raisons expliquées au chapitre 4.

Dans la figure 2.4, où chaque note est distante de sa voisine d'un demi-ton, nous pouvons distinguer la même relation entre les intervalles: ton,ton,demi-ton,ton,ton,ton,demi-ton. Ceci signifie que nous pouvons jouer n'importe quelle gamme en choisisant une note de départ et en ajoutant les intervalles.

Il nous faut une méthode pour distinguer une gamme d'une autre; ceci est réalisé par l'armature d'une clef avec un certain nombre de dièses ou de bémols à côté de la clef, en début de portée. Ils sont placés sur la portée dans un certain ordre comme le montre la figure 2.5. Les armatures indiquent au musicien que chaque note placée au même endroit doit être jouée un demi-ton au-dessus si c'est un dièse, ou un demi-ton en dessous si c'est un bémol. Une portée sans dièse ni bémol est appelée portée en clef d'UT.

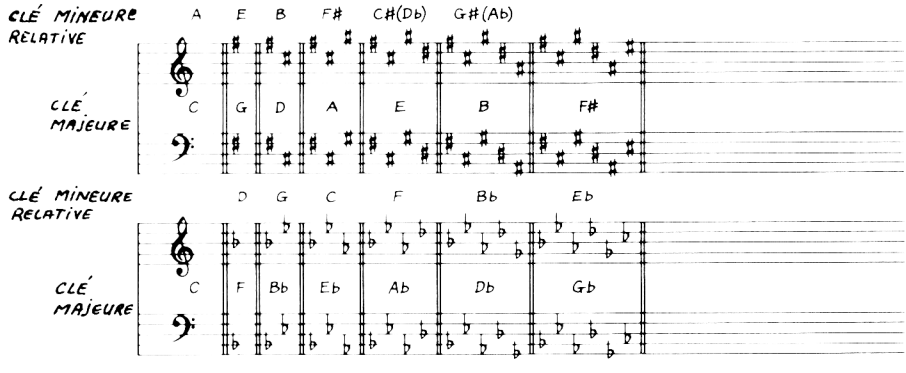


Figure 2.5. Les armatures.

Par exemple une gamme de RE avec deux dièses nous précise de jouer les FA et DO un demi-ton plus élevé. En nous référant à la figure 2.4 nous nous apercevons que cela produit un FA et un DO , et si nous jouons cette gamme, nous passons aux intervalles requis pour produire une gamme majeure. Dans un cas similaire la clef de FA indique que la note SI se transforme en SI bémol, pour produire la gamme de FA.

La gamme mineure

Les gammes forment les éléments de base pour composer des mélodies. Sur un clavier il est possible de jouer des gammes autres que les gammes majeures. Le fait de jouer uniquement les touches noires d'un piano nous donne une gamme de cinq notes, appelée gamme pentatonique. La ressemblance avec la musique orientale est très grande, du moins pour les occidentaux. On retrouve également cette sonorité chez tous les peuples ayant une culture musicale très ancienne: Chine, Polynésie... La gamme normale utilisant sept notes, les touches blanches du piano, est appelée diatonique et celle utilisant les douze notes: chromatique.

La gamme mineure est utilisée plus fréquemment. Il y a deux techniques pour former une gamme mineure: la mélodique et l'harmonique. Les deux formes ont la même armature de clef comme nous le montre la figure 2.5. La différence se trouve dans l'interprétation. Une gamme mineure harmonique ascendante passe par les intervalles suivants: ton,demi-ton,ton,ton,demi-ton,trois demi-tons,demi-ton. Si la gamme est prise en sens inverse les mêmes notes sont utilisées.

La gamme mineure mélodique est différente, quand vous jouez la gamme mineure ascendante la séquence est la suivante: ton,demi-ton,ton,ton, ton,ton,demi-ton. En descendant la séquence diffère: ton,demi-ton,ton,ton,demi-ton,ton. Vous n'êtes pas vraiment concernés par tout cela si

vous voulez uniquement reproduire de la musique sur l'AMSTRAD, mais sachez que cela existe et qu'une composition musicale peut inclure différentes formes de gammes.

Nous allons arrêter l'examen des gammes à ce niveau, avant de nous enliser dans le sujet. Il existe d'autres livres pour vous fournir des explications détaillées, mon propos n'étant que de vous familiariser avec les bases de la musique.

Quand une gamme majeure et mineure partagent la même armature, elle donne naissance à des tons relatifs. Par exemple LA mineur est le relatif mineur de DO majeur et FA est le relatif mineur de LA majeur.

Les gammes de DO majeur et de DO mineur sont illustrées un peu plus loin dans ce chapitre, en figure 2.11. Ainsi vous pouvez comparer les notes de la gamme avec les notes utilisées pour les accords.

Il existe d'autres gammes contenant un nombre variable de notes et d'intervalles, la plupart d'entres elles sont écrites en utilisant la notation standard, mais elles ne sont abordées qu'au cours de discussions académiques sur la théorie musicale.

Les notes enharmoniques

Pour être plus complet, il faut ajouter qu'une même note peut avoir deux noms différents. Par exemple un LA bémol est identique à un SOL , ou un MI bémol à un RE . Ces notes sont connues sous le nom de notes enharmoniques ou de notes synonymes.

Les accidents

Il peut arriver que vous vouliez jouer dans une certaine clef et qu'une note ne fasse pas

partie de cette clef. Vous pouvez établir momentanément une tonalité autre que celle de l'armature en plaçant un dièse ou bémol juste devant la note à altérer. Cette altération n'a d'effet que pour cette note précise. Il existe cinq signes d'altérations: (dièse) pour augmenter une note d'un demi-ton, le b(bémol) pour diminuer d'un demi-ton, le X(double dièse) et le bb(double bémol) qui respectivement augmente ou diminue de deux demi-tons. Il existe également une altération de précaution, (à) (1), qui supprime l'altération précédente et qui peut même être composée (àb) ou (à) ou encore (àà) pour supprimer totalement ou en partie les effets de (bb) ou de (XX).

La durée d'une note

Il y a deux aspects à ne pas confondre dans l'interprétation de la musique: la durée individuelle de chaque note et le rythme de la composition.

La durée d'une note est relative uniquement aux notes et en aucun cas elle ne détermine le rythme d'une pièce de musique. La durée des notes, dans la notation conventionnelle est donnée par la figure 2.6, sous les notations anglaise et américaine. La notation américaine est plus facilement compréhensible et nettement plus attrayante.

La valeur de la durée est toujours donnée par rapport à une autre note. Si une note est suivie d'un point, juste placé derrière elle, sa durée est augmentée de moitié. Une telle note est appelée note pointée. La mesure d'une pièce de musique est signalée par des indications en début de portée. Des interprétations rapides contiennent souvent des doubles, triples, voire quadruples croches et il est difficile de déterminer rigoureusement la mesure d'un morceau

(1) (à) correspond à un signe particulier similaire au dièse.

Notation	Nom français	Nom anglais	Nom américain	Valeur durée
𝅝	Ronde	Semibreve	Whole note	32
𝅗𝅥.	Blanche pointée	Dotted minim	Dotted half note	24
𝅗𝅥	Blanche	Minim	Half note	16
♩.	Noire pointée	Dotted crotchet	Dotted quarter note	12
♩	Noire	Crotchet	Quarter note	8
♪.	Croche pointée	Dotted quaver	Dotted eighth note	6
♪	Croche	Quaver	Eighth note	4
𝅘𝅥𝅯.	Double-croche pointée	Dotted semiquaver	Dotted sixteenth note	3
𝅘𝅥𝅯	Double-croche	Semiquaver	Sixteenth note	2
𝅘𝅥𝅰	Triple croche	Demi-semiquaver	Thirtysecond note	1

Figure 2.6. Notation anglaise et américaine des notes.

avec la notation comme seule indication.

Les silences jouent un important rôle et leurs valeurs sont indiquées en figure 2.7. Ici également nous trouvons tout un étagement des silences, la pause, la demi-pause, le soupir, le demi-soupir... jusqu'au seizième de soupir. En toute théorie, les silences peuvent également être augmentés de moitié par l'adjonction d'un point, mais cette notation est très peu usitée.

Notation	*Durée*
𝄻	32
𝄼	16
𝄽	8
𝄾	4
𝄿	2
𝅀	1

Figure 2.7. Les valeurs de silence.

La mesure

La mesure d'une pièce de musique est indiquée au début de la portée, directement après la clef, par deux chiffres l'un au-dessus de l'autre.

Le chiffre du haut indique le nombre de temps, le chiffre du bas la valeur de chaque temps. Par exemple 2/4 nous indique une mesure à deux temps, la valeur de chaque temps étant une noire. 3/4 nous donne une interprétation à trois temps avec une unité de temps toujours égale à une noire, c'est d'ailleurs la mesure utilisée dans la plupart des valses. Il est d'usage de remplacer 4/4 par un grand C et 2/2 par un grand C barré verticalement. Le C étant l'abréviation de commun, il est vrai que ce genre de mesure est très utilisé dans les fox-trot, les rock'n'roll, les ballades et la plupart des compositions classiques.

La mesure peut être changée à n'importe quel moment de la pièce musicale, en plaçant de nouveaux chiffres indicateurs précédés ou non d'une double barre. Ces différentes numérotations sont données en figure 2.8. En pratique ces indications devraient vous suffire, il n'y a que dans la musique d'avant-garde ou de jazz que l'on trouve des exceptions.

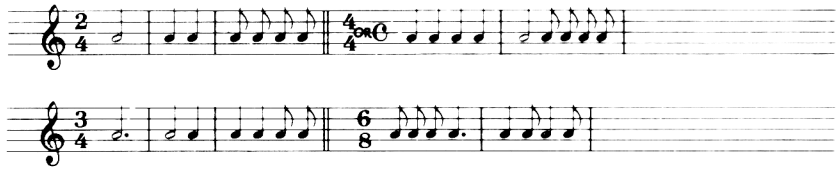



Figure 2.8. Les mesures simples et composées.

La liaison des notes

Vous pouvez très bien tomber sur l'un de ces aspects de la musique, aussi est-il important de les expliquer. Un triolet est un regroupement

de trois notes comme en montre la figure 2.9 ; elles sont jouées en même temps que deux notes de même valeur. Si vous voulez essayer, battez au même moment deux temps avec votre main gauche et six avec votre main droite. C'est très facile... Maintenant essayez de taper quatre temps avec votre main gauche et six avec votre main droite, c'est encore plus difficile. Pour l'ordinateur le problème est moins complexe, pour lui il ne s'agit que d'un problème de programmation et non de coordination.

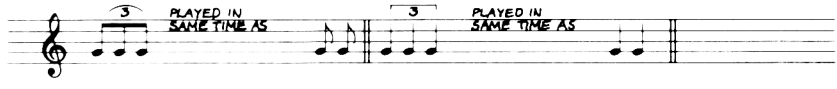



Figure 2.9. Les triolets.

Lorsque plusieurs de ces figures se succèdent, on remplace le plus souvent chacun des crochets par une barre horizontale commune. Il faut tenir compte de la division des temps dans l'emploi de ces barres. Il n'est pas indifférent, pour noter six croches, de mettre la barre horizontale sur les six, ou de dessiner deux fois une barre sur chaque groupe de trois croches, ou de faire trois groupes de deux croches.

La liaison est un signe courbe qui permet de réunir deux notes de même hauteur, sans répéter la deuxième (voir figure 2.10). Lorsque deux notes qui n'ont pas la même hauteur sont liées, il faut les jouer le plus en douceur possible.

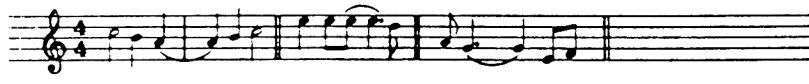



Figure 2.10. Les liaisons.

A l'opposé de la liaison nous avons le staccato. C'est un point placé au-dessus de la note, ou point suscrit, pour indiquer qu'il faut la séparer nettement de la note suivante et non "tenir" le son.

Harmonie et accords

On appelle harmonie le son qui résulte de plusieurs notes jouées simultanément (à l'opposé se trouve la mélodie qui naît de plusieurs notes jouées consécutivement). Si nous nous limitons à une octave seulement nous pouvons déjà combiner treize notes pour créer des milliers de combinaisons dont la plupart seront dissonantes. Même en nous limitant à trois ou quatre notes à la fois, il y a moyen de créer de nombreuses consonances. A travers les âges, certaines combinaisons ont fait leurs preuves et forment la base de nos mélodies actuelles. Ces combinaisons de notes sont appelées accords et en général elles sont le fruit de trois notes ou plus, ce qui en limite la production par l'AMSTRAD qui n'a que trois canaux.

Un accord est construit avec une séquence d'intervalles, un peu comme une gamme. L'accord le plus commun est l'accord majeur qui est basé sur des notes majeures.

Pour construire un accord en DO majeur nous ajouterons une tierce et une quinte à la note DO. Comme les intervalles, l'accord se lit de bas en haut, mais les différents intervalles qui le composent sont tous comptés à partir de la note inférieure appelée note de base.

Dans notre exemple la note de base est le DO; pour arriver à une tierce dans la gamme DO il faut ajouter un MI et pour la quinte un SOL (voir figure 2.4). Pour un accord mineur, la tierce tombe sur un bémol, donc un demi-ton plus bas. Ceci a pour effet de produire un son plus triste que le son réalisé par un accord majeur.

Les accords portent souvent le nom de leur construction et l'on peut trouver des noms comme: accord de neuvième majeur ou accord de neuvième sur septième mineur (voir les exemples donnés en figure 2.11).

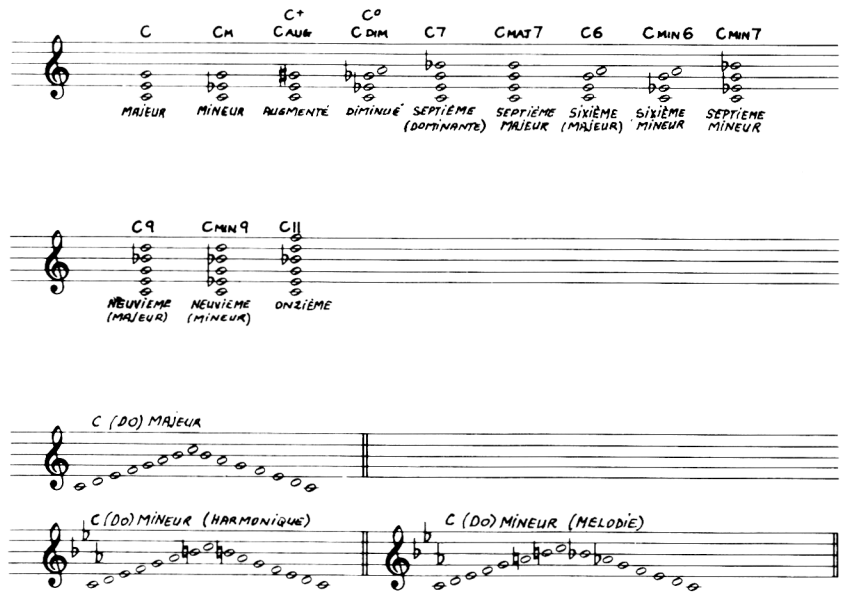


Figure 2.11. Les accords et les gammes.

Les accords sont très utiles et nous les expérimenterons un peu plus loin dans ce livre. En attendant, si vous voulez écouter les sons produits par les différents accords, essayez le programme suivant:

```
100 REM PROGRAM 2.1
110 REM Démonstration d'accords
120 :
130 MODE 2:PRINT:PRINT:PRINT
140 :
150 FOR accord=1 TO 4
160 READ accord$,note$,hauteur1,hauteur2
170 PRINT"C ";accord$;" = C + ";note$
180 SOUND 1,478,300
190 SOUND 2,hauteur1,300
200 SOUND 4,hauteur2,300
210 FOR d=1 TO 3000:NEXT d
220 NEXT accord
230 :
240 REM Accords à 4 notes
250 FOR arp=1 TO 5
260 READ accord$,note$
270 PRINT"C ";accord$;" (Arpège) = C +";
    note$
```

```
280 FOR note=1 TO 17
290 READ hauteur
300 SOUND 1,hauteur,10
310 NEXT note
320 FOR d=1 TO 1000:NEXT d
330 NEXT arp
340 END
350 :
360 DATA Majeur,E + G,379,319
370 DATA Mineur,Eb + G,402,319
380 DATA Augmenté,E + G#,379,301
390 DATA Suspendu 4ème,F + G,358,319
400 DATA Diminué, Eb +Gb + A,478,402,338
    ,284,239,201,169,142,119,142,169,201
    ,239,284,338,402,478
410 DATA 7ème,E+ G + Bb,478,379,319,268,
    239,190,159,134,119,134,159,190,239,
    268,319,379,478
420 DATA Majeur 7ème,E + G + B,478,379,
    319,253,239,190,159,127,119,127,159,
    190,239,253,319,379,478
430 DATA Majeur 6ème,E + G + A,478,379,
    319,284,239,190,159,142,119,142,159,
    190,239,284,319,379,478
440 DATA Mineur 6ème,Eb + G + A,478,402,
    319,284,239,201,159,142,119,142,159,
    201,239,284,319,402,478
```

Comme nous sommes limités à trois canaux, les accords contenant plus de trois notes sont joués en arpège, ainsi qu'on appelle le fait de jouer les notes en rapide succession au lieu de les jouer toutes ensemble.

Il y a d'autres types d'accord, mais les règles de construction restent les mêmes. Maintenant que nous sommes armés de toutes les connaissances acquises aux deux derniers chapitres, nous allons voir l'application pratique au niveau de l'AMSTRAD, avec la commande SOUND.

3

La commande SOUND

Quand l'interpréteur Basic rencontre la commande SOUND, il passe le contenu des paramètres au synthétiseur. Les commandes ENV et ENT n'affectent simplement que le volume, les caractéristiques de la fréquence d'une note n'entrant pas en jeu, du moins dans un premier temps.

Dans la forme la plus simple, l'instruction SOUND ne demande que les deux premiers paramètres, mais elle peut être affectée de sept paramètres différents. La manière la plus sûre de programmer la commande SOUND est de n'ajouter les cinq derniers paramètres qu'en cas de nécessité.

Vous n'êtes pas obligé d'engloutir dès la première lecture toutes les informations, vous pouvez très bien revenir à ce chapitre en cas de problème. Des programmes de démonstration sont inclus pour faciliter les explications; nous utiliserons toutes les commandes et les paramètres cités pour voir leur application pratique.

Dans le guide d'utilisation, la commande SOUND est décrite avec une série de parenthèses ou de crochets pour indiquer les paramètres optionnels, ce que nous éviterons ici pour plus de clarté. Voici sa syntaxe:

```
SOUND G,H,I,J,K,L,M
```

et voici la signification de chaque paramètre:

G = canal
H = hauteur
I = durée
J = volume
K = enveloppe du volume
L = enveloppe de tonalité
M = fréquence du bruit

Cette notation n'ayant vraiment rien de mnémonique je propose la notation suivante qui me semble plus facile:

SOUND C,P,D,V,VE,TE,N

C = canal
P = hauteur (anglais "pitch")
D = durée
V = volume
VE = enveloppe du volume
TE = enveloppe de tonalité
N = fréquence du bruit (anglais "noise")

Hauteur ou fréquence de ton

Il semble également que le terme de "hauteur" soit plus compréhensible que le terme de "fréquence de ton" avancé dans le guide d'utilisation, aussi l'emploierons-nous. Quant au terme "enveloppe de tonalité", nous le garderons, tout en comprenant bien qu'il ne se réfère en aucune manière à la forme de l'onde mais seulement à son amplitude.

File d'attente et mémoire-tampon

Quand une commande SOUND est décodée, les paramètres sont placés dans une file d'attente et le Basic reprend immédiatement la main. Chaque canal a son propre tampon et peut contenir jusqu'à quatre notes alors qu'une cinquième est jouée. Si la mémoire-tampon est pleine, la commande SOUND ne peut être interprétée et le programme Basic se met en attente.

En figure 3.1 nous trouvons une liste de référence concernant les paramètres de la commande SOUND, avec les limites et la valeur de chaque paramètre. Nous allons passer en revue les différents attributs.

Paramètre	Nom	Spectre	Défaut
C	Canal	1 à 255	aucun
P	Hauteur	0 à 4 095	aucun
D	Durée	– 32 768 à 32 767	20
V	Volume	0 à 15 (voir texte)	4 (lorsqu'il n'y a pas d'ENV)
VE	Enveloppe de volume	0 à 15	0
TE	Enveloppe de tonalité	0 à 15	0
N	Bruit	0 à 31	0

Figure 3.1. Les paramètres de la commande SOUND et leur échelle de valeur.

C = le canal

Ce paramètre peut prendre une valeur quelconque entre 1 et 255. Il n'y a pas de valeur par défaut, et un nombre doit obligatoirement être affecté à C. Le générateur de son contient trois canaux, et le numéro du canal est indiqué par ce paramètre. Pour spécifier un canal nous n'utiliserons pas les valeurs 1, 2 ou 3 comme vous auriez pu vous y attendre, mais plutôt 1, 2 et 4. Le guide d'utilisation se sert de la numérotation A, B et C, ce qui ne fait que masquer le problème et n'arrange en rien la compréhension; en effet, la valeur de ce paramètre étant dirigée sur un ordinateur qui ne sait interpréter que des nombres binaires, il faudra s'adapter à son langage. Un petit coup d'oeil au système binaire facilitera sûrement la compréhension.

Le système binaire

Nous, les humains, utilisons un système de calcul basé sur dix caractères, c'est-à-dire qu'avec un seul symbole nous savons interpréter dix valeurs différentes; un chiffre peut en effet prendre les valeurs 0 à 9. Les

ordinateurs ont une logique basée sur des signaux électriques et ne peuvent faire la différence qu'entre deux états. Il serait en effet trop compliqué de vouloir différencier dix états sur un même fil, on se limite donc à vérifier s'il y a ou non présence d'un courant. Ce système basé sur deux états est appelé système binaire. Le mot "bit", dérivé de l'anglais BInary digiT, représente une information électrique. On regroupe généralement 8 bits pour former un nombre ou un caractère. Le caractère W ou le nombre 137 occupent en mémoire 8 bits ou 1 octet (appelé en anglais "byte".) L'AMSTRAD est un micro-ordinateur 8 bits, ce qui signifie que tous les échanges entre le micro, la mémoire, et le générateur de son se font sur la base d'un octet.

Si tous les bits d'un octet sont à 1, soit 11111111, il prend la valeur 255; si tous les bits sont à 0, soit 00000000, il prend la valeur zéro. Si vous n'êtes pas familiarisé avec le comptage binaire, lisez la suite.

Dans notre système décimal, quand nous déplaçons un chiffre d'un rang, ce chiffre prend une valeur dix fois supérieure à celle qu'il avait dans son ancienne position à droite. Par exemple:

26 425 = 5x1 + 2x10 + 4x100 + 6x1 000 + 2x10 000

Dans le système binaire, à chaque fois que nous passons d'un rang à un autre le multiplicateur n'est plus de 10 mais de 2. Ainsi:

10110110 = 0x1 + 1x2 + 1x4 + 0x8 + 1x16 + 1x32
+ 0x64 + 1x128

Nous appelons les rangs décimaux dix, cent, mille... et nous passons de rang en rang en incrémentant la valeur de base des valeurs successives des puissances de 2, de la manière suivante:

10^4	10^3	10^2	10^1	10^0
10 000	1 000	100	10	1
2	6	4	2	5

Voici comment nous procédons en système binaire:

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

Vous devez être capable maintenant de pouvoir interpréter des nombres binaires. Par exemple 11111111 est équivalent à 255 en décimal. Vous ne pouviez affecter une valeur supérieure à 255 en utilisant l'instruction POKE, parce que la valeur maximale d'un octet est 255.

Pour différencier les 8 bits d'un octet, on les numérote de 0 à 7 en partant de la droite. Le bit de droite représente le poids binaire le plus faible, il est appelé LSB (Least Significant Bit). Celui de gauche par contre représente le poids le plus fort, c'est le MSB (Most Significant Bit).

Le système binaire est à la base de tous les systèmes informatiques, il est donc important d'être bien familiarisé avec ce calcul. L'annexe 2 du guide d'utilisation approfondit l'étude du calcul binaire. Le Basic de l'AMSTRAD est d'une utilisation très facile et nous n'avons que peu d'occasions de programmer au niveau du bit. Il en va différemment des commandes du synthétiseur...

Retournons à notre programmation du canal dans la commande SOUND. La manière de programmer est visualisée en figure 3.2. Pour envoyer un son vers les canaux A et C nous programmons les bits 0 et 2, soit 00000101 en binaire et 5 en décimal. En attendant d'être plus familiarisé avec la représentation binaire vous pouvez utiliser la figure 3.2 pour vos

translations. En Basic, on spécifie une valeur binaire en préfixant le nombre avec le signe &h. Pour envoyer un DO vers les canaux A et C il faut écrire:

Bit	Décimal	Commande
0 LSB	1	Envoyer le son vers le canal A
1	2	Envoyer le son vers le canal B
2	4	Envoyer le son vers le canal C
3	8	Rendez-vous avec le canal A
4	16	Rendez-vous avec le canal B
5	32	Rendez-vous avec le canal C
6	64	Retenue
7 MSB	128	Vidage

Figure 3.2. Statut des chaînes - explication de la correspondance des bits.

```
SOUND &X101,478
```

Le rendezvous

L'option **rendezvous** permet à des notes sur différents canaux de se synchroniser et de produire leur son au même moment. Par exemple, pour synchroniser A et B, un bit de **rendezvous** doit être placé sur A pour lui signifier de se synchroniser avec B , et il en est de même pour B. S'il n'y a aucun **rendezvous** de placé, les notes sur les différents canaux sont produites dès leur arrivée en tête de la file d'attente.

```
100 REM PROGRAM 3.1
110 REM Démonstration de rendezvous
120 :
130 REM Sans rendezvous
140 SOUND 1,478,200
150 FOR d=1 TO 1000:NEXT d
160 SOUND 2,379,200
170 SOUND 4,319,200
180 :
190 GOSUB 410
200 :
210 REM Rendezvous
```

```
220 REM A avec B & C
230 REM B avec A & C
240 REM C avec A & B
250 SOUND 49,239,200
260 FOR d=1 TO 1000:NEXT d
270 SOUND 42,190,200
280 SOUND 28,159,200
290 :
300 GOSUB 410
310 :
320 REM Rendezvous de A & B & C
330 rend=56
340 SOUND 1+rend,119,200
350 FOR d=1 TO 1000:NEXT d
360 SOUND 2+rend,95,200
370 SOUND 4+rend,80,200
380 END
390 :
400 REM Attendre de presser une touche
410 PRINT"Presser la touche"
420 k$="":WHILE k$="":k$=INKEY$:WEND
430 PRINT"D'accord"
440 RETURN
```

Si le Basic est en train de travailler sur les données, alors un léger délai est introduit entre les deux commandes SOUND. La ligne 150 introduit un délai pour illustrer la manière de travailler sans synchronisation. Le second exemple utilise l'option **rendezvous** pour synchroniser deux canaux. Pour placer une synchronisation sur les trois canaux nous pouvons utiliser un **rendezvous** en programmant le paramètre C à 56 ou &h111000. Naturellement, chaque canal peut avoir un rendez-vous avec lui-même. Si plus d'un canal est spécifié, alors le rendez-vous a lieu automatiquement.

Quand nous programmons une suite de commandes SOUND, nous ne recherchons pas toujours la synchronisation. Un canal peut servir à plusieurs notes alors que l'autre ne jouera que deux tons. Si les premières notes ont été synchronisées, les autres notes seront jouées en mesure, aussi longtemps que le Basic n'est pas freiné par des interruptions. Nous verrons tout ceci en détail au Chapitre 9.

Les commandes HOLD et RELEASE

"Hold" veut dire "retenir", et "release" "relâcher".

Un HOLD est mis en place avec le bit 5. Il a pour effet de maintenir le son dans la file d'attente. L'effet de la commande HOLD est supprimé lorsqu'on vide les canaux ou que l'on utilise la commande RELEASE affectée au bit 7.

La commande HOLD peut être utilisée comme alternative à l'option **rendezvous** pour démarrer tous les canaux au même instant. Pour voir exactement la manière dont travaillent ces bits nous allons entrer le programme suivant:

```
100 REM PROGRAM 3.2
110 REM HOLD & RELEASE:Démonstration
120 :
130 REM ON BREAK GOSUB 230
140 :
150 FOR gamme=1 TO 8
160 IF gamme=1 THEN statut=65 ELSE
    statut=1
170 READ note
180 SOUND statut,note
190 NEXT gamme
200 END
210 DATA 478,426,379,358,319,284,253,239
220 :
230 ON BREAK STOP
240 PRINT gamme
250 PRINT"Presser une touche pour le
    RELEASE"
260 k$=INKEY$:IF k$="" GOTO 260
270 RELEASE 1
280 RETURN
```

Commentaires

Ce programme joue, ou essaye de jouer, une simple gamme. La ligne 160 met en place un paramètre HOLD sur la première note de la gamme uniquement. Au premier RUN rien ne semble se passer. Appuyez sur la touche ESC deux fois et

vous allez voir:

```
BREAK en 180
ENTER
PRINT gamme
```

et un 5 sera affiché pour montrer que la commande SOUND a généré quatre sons avant d'être interrompue dans la génération du cinquième. Otez le REM dans l'instruction 130. Faites RUN à nouveau et appuyez sur ESC deux fois de suite. Le programme sera dérouté vers la ligne 230. La ligne 240 affiche la valeur **gamme** qui sera à nouveau 5. Appuyez sur n'importe quelle touche pour émettre un RELEASE 1 en ligne 270. Les quatre notes gelées dans la file d'attente sont libérées et jouées. Le programme revient en ligne 280 puis 180 pour les notes suivantes.

Le paramètre RELEASE se programme également au niveau du bit: 1, pour le canal A, 2 pour le B, 4 pour C ou 7 pour les 3 canaux. Il n'a d'effet que si un canal est en HOLD.

En pratique, pour synchroniser les trois canaux un HOLD va être émis avec:

```
SOUND 71,0,1,0
```

Pourquoi 71?

```
N  bit =  7  6  5  4  3  2  1  0

Valeur= 128 64 32 16  8  4  2  1
          ù  ù  ù  ù  ù  ù  ù  ù
          ù  ù  ù  ù  ù  ù  ù  ù------  CANAL A
          ù  ù  ù  ù  ù  ù  ù---------  CANAL B
          ù  ù  ù  ù  ù  ù------------  CANAL C
          ù  ù  ù  ù  ù---------  RENDEZ-VOUS A
          ù  ù  ù  ù------------  RENDEZ-VOUS B
          ù  ù  ù---------------  RENDEZ-VOUS C
          ù  ù---------------------------  HOLD
          ù---------------------------  RELEASE
```

Si le bit 7 est programmé à 1, le segment du canal est vidé et l'émission de la note est

stoppée. Puis la commande SOUND contenant la commande RELEASE 4 est jouée. Cet effet est utilisé dans les jeux pour passer d'un fond musical à une tonalité différente, pour un déplacement d'objet par exemple. Ce passage d'une tonalité à une autre peut se faire instantanément, et le bruit des explosions peut être synchrone avec les effets graphiques.

En utilisant le Programme 3.2 nous pouvons illustrer la commande RELEASE en altérant la ligne 270:

```
270 SOUND 129,0,1,0
```

La ligne 130 n'est toujours pas passée en commentaire (en REM). Cette fois-ci, quand le programme va rentrer dans la routine ON BREAK, le canal va se vider de ses quatre notes. Retour à la ligne 180 pour jouer les quatre notes restantes.

Dans un contexte musical, il peut être sage d'inclure un paramètre RELEASE avant un HOLD pour vider le segment de toutes les notes restantes.

```
SOUND 199,0,1,0
```

L'analyse du paramètre C de la commande SOUND de l'AMSTRAD nous montre la souplesse d'utilisation de ce micro et son adaptation aux différents contextes. Les autres paramètres sont généralement d'usage plus simple.

P = la hauteur

Cette valeur doit être comprise entre 0 et 4095 et il n'existe aucune valeur par défaut. Elle détermine la hauteur ou fréquence du son. La figure 2.4, dans l'annexe 1, liste les hauteurs requises pour produire les notes usuelles de notre gamme occidentale. Sa valeur n'est qu'une convention interne à l'AMSTRAD, pour faire vibrer le haut-parleur à la fréquence

de la note. Vous pouvez programmer n'importe quelle valeur comprise entre les limites spécifiées ci-dessus mais vous remarquerez que certaines valeurs produisent des sons orientaux. Nous verrons dans le chapitre suivant comment calculer la hauteur des notes situées en dehors du tableau.

Le guide d'utilisation spécifie, au chapitre 6, qu'aucune note n'est produite si la fréquence sélectionnée est 0. En fait, un petit "clic" se fait entendre. Essayez le programme suivant:

```
10 FOR hauteur=10 TO 0 STEP -1
20 SOUND 1,hauteur
30 NEXT hauteur
```

Il est suggéré de n'utiliser la valeur 0 que dans le cas où l'on travaille avec l'option NOISE ("bruit"); dans ce cas, le petit "clic" est largement étouffé.

D = la durée

Ses limites sont -32 768 et 32 767, sa valeur par défaut est de 20. Si la valeur programmée est positive et supérieure à 0, la durée est programmée en 1/100 de seconde. Si la valeur égale 0, la durée est pilotée par la longueur de l'enveloppe de volume. Par contre, si aucune enveloppe de volume n'est définie, comme dans l'exemple suivant:

```
SOUND 1,478,0
```

alors le paramètre D est égal à la valeur par défaut de l'instruction ENV, soit 200.

Si la durée est inférieure à zéro, valeur absolue, le nombre, sans tenir compte du signe, stipule le nombre de répétitions de l'enveloppe de volume. Par exemple ces deux sons ont la même durée:

```
SOUND 1,478,400
SOUND 1,478,-2
```

La valeur par défaut de ENV est 200.

Ceci nous permet déjà un contrôle plus important du synthétiseur et nous l'utiliserons surtout quand des commandes ENV très compliquées seront utilisées.

V = le volume

La gamme va de 0 à 15. Si aucune ENV n'est spécifiée, la valeur se limite de 0 à 7, les valeurs 8 à 15 n'étant qu'une répétition de la première gamme. Si ENV est défini, le volume peut prendre une des seize valeurs.

Le guide d'utilisation stipule que la valeur par défaut est 12 si une ENV est définie, mais généralement cette instruction n'est définie qu'ultérieurement. L'intention du guide était probablement de dire qu'un volume de 4 sans ENV était identique à un volume de 12 avec ENV. Le paramètre V représente le volume initial, mais peut être altéré par ENV. Vous pouvez le tester par l'exemple suivant:

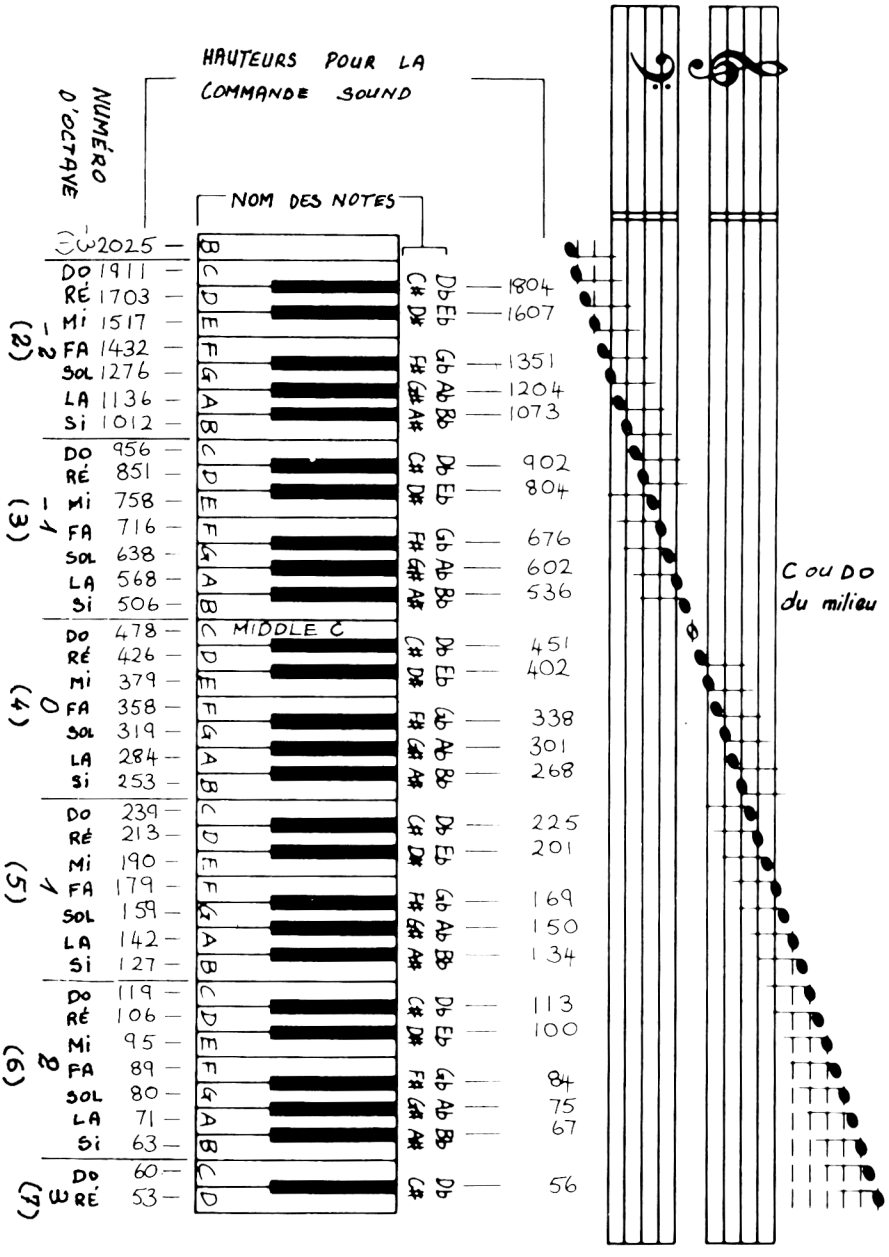
```
10 FOR vol = 0 to 15
20 SOUND 1,478,50,vol
30 NEXT vol
```

puis:

```
 5 ENV 1,1,15,1
10 FOR vol = 0 to 15
20 SOUND 1,478,50,vol,1
30 NEXT vol
```

VE = l'enveloppe de volume

Sa valeur par défaut est de 0, ses limites sont 0 et 15. Le paramètre VE spécifie le numéro de l'enveloppe de volume qui contrôlera l'émission du son. ENV doit naturellement être


HAUTEURS POUR LA COMMANDE SOUND
NUMÉRO D'OCTAVE
NOM DES NOTES
MIDDLE C
C ou DO du milieu
DO 2025
DO 1911
RÉ 1703
Mi 1517
FA 1432
SOL 1276
LA 1136
Si 1012
DO 956
RÉ 851
Mi 758
FA 716
SOL 638
LA 568
Si 506
DO 478
RÉ 426
Mi 379
FA 358
SOL 319
LA 284
Si 253
DO 239
RÉ 213
Mi 190
FA 179
SOL 159
LA 142
Si 127
DO 119
RÉ 106
Mi 95
FA 89
SOL 80
LA 71
Si 63
DO 60
RÉ 53
(2) -2
(3) -1
(4) 0
(5) 1
(6) 2
(7) 3
Db Eb Gb Ab Bb
C# D# F# G# A#
1804
1607
1351
1204
1073
902
804
676
602
536
451
402
338
301
268
225
201
169
150
134
113
100
84
75
67
56

défini auparavant. Si ce n'est pas le cas, alors la valeur par défaut 0 sera utilisée. De plus, ENV 0 ne peut être redéfini par l'utilisateur.

Le volume d'un son peut être modifié par plusieurs facteurs, comme nous l'avons vu précédemment, au Chapitre 1. Si l'amplitude d'une fréquence est doublée, elle n'est pas forcément perçue deux fois plus forte. Ceci pour des raisons physiologiques, car nous percevons les sons suivant une courbe logarithmique. Si un vibrato ou un trémolo est appliqué à une note, elle paraîtra plus forte. Le volume varie également avec la fréquence et il faut appliquer une puissance supérieure aux notes basses pour qu'elles semblent aussi fortes que les aig es.

TE = l'enveloppe de tonalité

Ses limites sont identiques à celles de VE: 0 et 15, ainsi que sa valeur par défaut: 0. TE spécifie le numéro de l'enveloppe de tonalité (ENT) qui contrôlera la fréquence du son. ENT doit être défini, sinon la valeur par défaut est utilisée. De même ENT 0 ne peut être touché par l'utilisateur. ENT 0 n'a aucun effet sur le son et il est émis à l'état brut.

N = le bruit

(En anglais: "noise".)

L'étendue de ses valeurs peut aller de 0 à 31. Une erreur s'est glissée dans le guide d'utilisation (version anglaise) qui parle d'une limitation à 15. Sa gamme correcte est donnée à la page F3.20. Sa valeur par défaut est 0 qui ne représente aucun son.

Ci-dessous, voici des exemples pour écouter les sons disponibles:

```
10 FOR n = 1 to 31
20 SOUND 1,0,200,7,0,0,n
30 Next n
```

Ce paramètre peut être utilisé pour produire tous les types de sons. Le synthétiseur ne peut générer qu'un seul son issu d'un paramètre N. Si plusieurs valeurs de N sont placées sur plus d'un canal, alors chaque nouvelle valeur détruit l'ancienne. Tous les canaux ayant un N actif produisent du bruit à la même fréquence.

Une sections entière sera consacrée à la génération des bruits, au chapitre 7.

Mettons-les tous ensemble

Tous les paramètres SOUND doivent être en valeurs entières. Si tel n'est pas le cas, les valeurs sont arrondies, d'une manière similaire à la fonction ROUND:

SOUND 1,16.499 produit le même effet que
SOUND 1,16

et

SOUND 1,16.5 est identique à
SOUND 1,17

SQ(c) = l'état de la file d'attente

Après avoir programmé un grand nombre de bruits complexes, il est souvent difficile de "retrouver ses petits" et savoir ce que fait chaque canal à un instant précis. La fonction SQ nous permet d'interroger un canal et de connaître son état, le nombre de places disponibles dans la file, HOLD ou non... La valeur (c) de SQ(c) désigne la canal à interroger . Les valeurs de c sont 1 pour A, 2 pour B et 4 pour C. En retour nous recevons un nombre entier, qu'il nous faut transformer en binaire, et qui représente le statut du canal. La translation peut se faire en utilisant la

fonction BIN$ du Basic (voir Chapitre 8 page 4 du guide d'utilisation, version anglaise), de la manière suivante:

```
PRINT BIN$(SQ(1))
```

Et voici la signification de l'information qui nous est retournée:

```
Bit 0 à 2: nombre binaire indiquant le nombre
           de places libres dans la queue
Bit 3    : rendez-vous canal A
Bit 4    : rendez-vous canal B
Bit 5    : rendez-vous canal C
Bit 6    : HOLD
Bit 7    : canal en train de jouer
```

Si un HOLD et un **rendezvous** sont mis en place sur le même canal, seule l'information du bit 6 sera retournée. Les bits 6 et 7 sont mutuellement exclusifs, nous ne pourrons jamais avoir les deux affichés ensemble.

```
100 REM PROGRAM 3.3
110 REM Démonstration SQ
120 :
130 rend=56
140 SOUND rend+1,478,200
150 GOSUB 210
160 SOUND rend+2,379,200
170 GOSUB 210
180 SOUND rend+4,319,200
190 GOSUB 210
200 END
210 PRINT "A...";BIN$(SQ(1),8)
220 PRINT "B...";BIN$(SQ(2),8)
230 PRINT "C...";BIN$(SQ(4),8)
240 PRINT
250 RETURN
```

Notez que le dernier **rendezvous** dans le canal C n'est pas retourné parce que, dès son émission, tous les canaux deviennent actifs et jouent. Pour voir l'effet d'un HOLD, altérez les lignes 180 et ajoutez la ligne 195.

```
180 SOUND rend+4+64,319,200
195 RELEASE 4:GOSUB 210
```

Quand une instruction SQ est générée, elle met hors-fonction le ON SQ(c) GOSUB, interruption mise en place pour ce canal. Nous allons le vérifier immédiatement.

ON SQ(c) GOSUB= Pour remplir les files d'attente

Le Basic AMSTRAD permet au programmeur de placer quelques interruptions (voir le guide d'utilisation, chapitre 10, page 1.) Une telle instruction placée en début de programme nous permet d'attendre un événement. S'il se produit, le cours normal du programme est dérouté vers la sous-routine spécifiée par le GOSUB. Le ON SQ(c) GOSUB est spécialement destiné à la production des sons. Le paramètre c est binaire et prend les 1, 2, 4. Une fois en place, le déroutage du cours normal ne se fera que s'il reste une place libre dans la file du canal spécifié. De cette manière, nous sommes sûrs de pouvoir approvisionner correctement notre synthétiseur.

Il y a quatre horloges d'interruption (numérotées de 0 à 3); elles ont des niveaux de priorité différents, 3 étant la plus grande et 0 la plus faible.

Les files d'attente du synthétiseur ont des interruptions indépendantes mais toutes de même niveau, équivalant au niveau d'interruption du minuteur 2. De cette manière, si une routine d'interruption a commencé, elle ne peut être interrompue par une autre.

Le fait d'interrompre fait disparaître l'événement; d'autre part, pour réarmer l'interruption il faut refaire ON SQ GOSUB. Il en est de même lorsqu'on programme la fonction SQ.

Les interruptions du générateur de son s'avèrent d'une grande utilité pour le

programmeur. C'est avec ces instructions qu'il peut assurer l'approvisionnement et le contrôle du synthétiseur, pendant que le Basic continue son travail. La programmation en multi-canaux, que nous utiliserons plus tard, en est nettement simplifiée.

Le son stéréo et les haut-parleurs extérieurs

Vous pouvez apporter un "plus" au son produit par votre AMSTRAD, en utilisant les possibilités stéréo de ce micro. Pour cela, il faut connecter le port d'entrée-sortie I/O, situé à l'arrière du micro, à une chaîne hi-fi ou à un casque stéréo. Le canal A assure la production d'une voie, le canal C l'autre, et le canal B un mélange des deux pour simuler le son venant du centre.

Un mini-jack stéréo est nécessaire; généralement les écouteurs du type "walkman" sont pourvus de ce genre de fiche. Dans le commerce, vous trouverez probablement des câbles pour connecter l'AMSTRAD à votre chaîne. Il y a deux avantages à passer par une chaîne: la puissance et la qualité musicale. Dans ce livre, la plupart des programmes produisant de la musique en multi-canaux utilisent l'effet stéréo.

L'effet panoramique (anglais: "panning")

Cet effet est obtenu en positionnant un son et en le déplaçant dans le champ stéréo. Il est très impressionnant de superposer une mobilité à un son. Vous avez ici une petite démonstration de cet effet:

```
10 ENV 1,1,15,1,10
20 ENV 2,1,15,-1,10
30 SOUND 1,956,0,0,1
40 SOUND 4,956,0,15,2
```

Il faudrait changer le dernier paramètre dans l'instruction ENV et nous verrons tout cela

au chapitre 5. L'effet panoramique est souvent utilisé dans la programmation du bruit des missiles se déplaçant d'un bout à l'autre de l'écran.

La commande SOUND par elle-même est déjà assez complète, mais l'étendue des sons qu'elle peut générer sera considérablement augmentée par les instructions ENV et ENT que nous examinerons en détail au Chapitre 5.

4

Programmer notes et gammes

Le synthétiseur de l'AMSTRAD peut produire des fréquences allant de 31 à 10 000 Hz. Avec un tel registre nous balayons largement le spectre des différentes tonalités, en allant d'une série de cliquetis à la simulation d'un piccolo. Nous allons explorer le potentiel de ce générateur au long de ce livre.

Nous pouvons programmer ce micro-ordinateur pour générer des gammes, autres que celles utilisées couramment. Certaines gammes,utilisées dans des musiques de style différent, ne peuvent être générées que par des synthétiseurs très coûteux ou alors des ordinateurs comme l'AMSTRAD.

Mais, tout d'abord, voyons d'où viennent les gammes.

Calculons la hauteur

Nous appellerons **hauteur** le nombre qu'il faudra entrer à l'AMSTRAD pour reproduire une fréquence associée à une note. Tout le monde connaît le "LA" du diapason qui a pour fréquence 440 Hz. Pour reproduire cette fréquence, il faut entrer au clavier une hauteur de 284.

Pour vous aider, vous trouverez, à l'annexe 1, un tableau mettant en relation la note, la fréquence, et la hauteur. Ces indications couvrent les notes des gammes normalement

utilisées sur les huit octaves du synthétiseur. Nous pouvons utiliser ces valeurs directement, sans information supplémentaire, mais pour les aventuriers et les curieux nous allons voir d'un peu plus près l'origine de ces nombres.

Si nous connaissons la fréquence en hertz de la note à reproduire, nous pouvons calculer la hauteur en utilisant la formule suivante:

hauteur=ROUND(125000/Hz)

D'un point de vue musical, cette formule ne veut rien dire, elle n'est là que pour permettre de programmer l'AMSTRAD pour produire un son bien précis. Le programme suivant nous fait une démonstration audible de la formule de hauteur. Il demande une fréquence en entrée, imprime la hauteur et joue la note:

```
100 REM PROGRAM 4.1
110 REM Conversion des hertz en hauteur
120 :
130 CLS
140 WHILE -1
150 INPUT "hertz";hz
160 hauteur=ROUND(125000/hz)
170 PRINT"hauteur =";hauteur:PRINT
180 SOUND 1,hauteur,100
190 WEND
```

Pour utiliser ces fréquences dans un contexte musical, nous devons être capable de produire des hauteurs spécifiques, ayant des relations musicales entre elles. L'exemple le plus évident est notre gamme occidentale. Il existe des gammes différentes de celle-ci et il est intéressant de voir comment la gamme occidentale est apparue.

La gamme tempérée

Les notes utilisées dans la musique occidentale forment ce que nous pouvons appeler des gammes tempérées. Ceci consiste en des

octaves, divisés en douze intervalles égaux appelés demi-tons. Le nom de "gamme tempérée" vient du fait que les notes ont été serrées, plus que les relations harmoniques ne l'auraient demandé. Le tempérament est le procédé qui consiste à réduire le nombre de sonorités par octave en altérant la fréquence des notes par rapport à leur fréquence naturelle.

Cette notion de fréquence naturelle peut sembler bizarre, ainsi que le fait de réduire le nombre de tons par octave. Il y a treize tons dans une octave, mais jadis elles étaient beaucoup plus nombreuses; on en comptait dix-sept exactement.

La gamme de « juste intonation »

On a pu remarquer que la combinaison la plus harmonieuse des tonalités est produite quand les fréquences ont un rapport mathématique simple entre elles. Par simple, j'entends 8/5 ou 5/4, non 1,189207/1.

Nous pouvons construire une gamme basée sur des intervalles de fréquence simples, connus sous le vocable de séries harmoniques; cela produira une gamme de "juste intonation". Une gamme basée sur ces séries harmoniques produira des notes plus plaisantes à écouter que celles produites par une gamme tempérée. Il semblerait donc évident d'utiliser les tonalités les plus harmonieuses, à savoir la gamme de "juste intonation".

L'usage d'intervalles de fréquence basés sur les séries harmoniques produit trois types d'intervalles que l'on dénomme tons majeurs, tons mineurs et demi-tons (alors que dans la gamme tempérée, un demi-ton est un ton mineur).

Pour clarifier tout cela, la figure 4.1 nous montre le ratio entre les intervalles dans une gamme de "juste intonation". Cette figure nous indique également les fréquences utilisées dans

une gamme de DO majeur. Dans la dernière colonne nous avons le ratio entre les notes adjacentes de la gamme: un ton majeur a un rapport de 9/8, un ton mineur un rapport de 10/9 et un demi-ton un rapport de 16/15. Par exemple, le rapport entre DO et RE est 9/8 et entre RE et MI: 10/9.

Intervalle	Ratio/Unisson	Note	Ratio entre deux notes adjacentes
Unisson	1:1	C	9:8
Demi-ton	16:15		
Ton mineur	10:9		
Ton majeur	9:8	D	10:9
Troisième mineur	6:5		
Troisième majeur	5:4	E	16:15
Quatrième parfait	4:3	F	9:8
Quatrième augmenté	45:32		
Cinquième diminué	64:45		
Cinquième parfait	3:2	G	10:9
Sixième mineur	8:5		
Sixième majeur	5:3	A	9:8
Septième mineur harmonique	7:4		
Septième mineur grave	16:9		
Septième mineur	9:5		
Septième majeur	15:8	B	16:15
Octave	2:1	C	

Figure 4.1. Rapport entre les intervalles de la gamme de juste intonation.

La gamme « désaccordée »

Nous nous apercevons très rapidement que l'usage de tels rapports nous empêche de contruire d'autres gammes que celle de DO. Essayez de construire une gamme en RE. Le premier intervalle dont nous avons besoin est un intervalle majeur de RE à MI dont le rapport est 9/8. Nous pouvons voir sur la figure que la différence entre RE et MI est un ton mineur, soit 10/9.

Une fois que vous avez saisi tout cela vous comprendrez que les instruments, pour rester accordés, vont devoir se limiter à jouer dans une seule clef. Pour jouer dans des clefs différentes, un autre jeu de fréquences basé sur les rapports précédents doit être utilisé.

Bien que cette gamme soit très harmonieuse, on a dû trouver une autre méthode, plus simple et plus flexible. On a divisé l'octave en 12 intervalles égaux, c'est la gamme tempérée que nous utilisons aujourd'hui. La gamme de Pythagore, avec ses intervalles purs, était très limitée dans le choix des clefs ainsi que dans les progressions harmoniques. La gamme tempérée a été adoptée universellement dans le courant du XVIIIe siècle. J.S. BACH a écrit une composition de 48 pièces: "Das Wohltemperierte Clavier", (Le Clavier Bien Tempéré), qui fait référence à ces gammes.

Pour comparaison, la figure 4.2 nous montre le rapport entre les intervalles dans une gamme tempérée . Si vous voulez expérimenter la gamme de "juste intonation", vous trouverez ci-dessous les fréquences en Hz de la gamme en DO octave 0:

Intervalle	Ratio/Unisson	Note
Unisson	1:1	C
Demi-ton/deuxième mineur	1.059463:1	
Ton/deuxième majeur	1.122462:1	D
Troisième mineur	1.189207:1	
Troisième majeur	1.259921:1	E
Quatrième parfait	1.334840:1	F
Quatrième augmenté / Cinquième diminué \	1.414214:1	
Cinquième parfait	1.498307:1	G
Sixième mineur	1.587401:1	
Sixième majeur	1.681793:1	A
Septième mineur	1.781797:1	
Septième majeur	1.887749:1	B
Octave	2:1	C

Figure 4.2. Rapport entre les intervalles de la gamme tempérée.

NOTE	Hz.
DO	264
RE	297
MI	330
FA	352
SOL	396
LA	440
SI	495
DO	528

Vous pouvez déterminer les autres clefs en utilisant les rapports de la figure 4.1. Par exemple, dans la clef de RE, pour arriver à la fréquence de la seconde note, MI, un intervalle majeur est requis, soit 9/8. Il faut donc multiplier par 9 et diviser par 8.

297/8*9 = 334,125

Vous pouvez vous apercevoir que la fréquence de ce MI est différente de la fréquence du MI en gamme de DO.

En complément, voici les fréquences de base des autres notes:

DO	278.4
REb	278.1
MIb	312.9
FA	371.2
SOLb	370.8
LAb	417.2
SIb	469.3
DOb	494.4

Si tout cela vous semble un peu confus, n'ayez aucune inquiétude. Partout dans ce livre nous produirons de la musique basée sur des gammes tempérées, c'est-à-dire la gamme classique. Avec l'augmentation de vos connaissances musicales, il vous arrivera peut-être de revenir à des conceptions de gammes avec des intervalles différents. C'est un sujet fascinant, un domaine grandiose d'exploration.

La gamme microtonale

La gamme d'égal tempérament divise l'octave en douze parties égales, mais nous pouvons essayer de la diviser d'une manière différente, en treize, quatorze parties ou plus. Les tonalités obtenues sont appelées microtonales. Rien n'est plus simple pour un ordinateur équipé d'un synthétiseur comme l'AMSTRAD; par contre, écrire des compositions basées sur ce genre de tonalités est une autre affaire... Nous reviendrons sur ce type de gamme d'ici quelques lignes.

Calcul des notes dans la gamme tempérée

Nous pouvons déterminer la fréquence des notes des gammes classiques en utilisant la formule suivante:

```
freq = 440* 2^(oct+(note-10)/12)
hauteur = ROUND(12500/freq)
```

où çOfreq est la fréquence en Hz, oct le numéro de l'octave, note le numéro de la note, c'est-à-dire 1 = DO, 2 = DO , 3 = RE... Ceci est la représentation des notes dans l'annexe 1. Notez que dans le guide d'utilisation, la formule de fréquence utilise l'expression "10-N" à la place de "N-10".

Vous remarquerez que toutes les notes sont calculées à partir du LA international, de 440 Hz. Même si vous n'êtes pas un amateur de musique averti, vous avez sûrement déjà entendu l'expression: "Donnez-moi le LA", utilisée pour accorder les instruments entre eux. Ceci garantit que tous les instruments sont accordés sur la même fréquence, pour pouvoir jouer ensemble.

La formule est basée sur le fait que chaque note d'une octave a une fréquence double par rapport à la même note dans l'octave inférieure. Vous avez pu observer, également, qu'à chaque

doublement de fréquence, la hauteur est divisée par deux. Cela semble contradictoire, mais aussi longtemps que nous garderons ceci présent à l'esprit, nous n'aurons pas de mauvaise surprise. L'expérimentation de cette formule se fera par le programme suivant:

```
100 REM PROGRAM 4.2
110 REM Octave et note
120 REM Conversion en hauteur
130 :
140 WHILE -1
150 INPUT "Octave";oct
160 INPUT "Note";note
170 freq=440*(2^(oct+(note-10)/12))
180 hauteur=ROUND(125000/freq)
190 PRINT"Freq =";freq,"hauteur =";
    hauteur
200 PRINT
210 SOUND 1,hauteur,100
220 WEND
```

Numérotation différente des notes et des octaves

Le guide d'utilisation repère l'octave du DO central par la valeur 0. Les octaves inférieures sont repérées par des signes négatifs, les octaves supérieures par des signes positifs. Cette méthode est souvent utilisée, mais elle n'est pas unique. Nous pouvons par exemple repérer l'octave la plus basse par 1 et n'employer que des nombres positifs. De même nous pouvons nous passer de la numérotation des octaves, en ne numérotant que les notes; le DO de l'octave 3 sera égal à 1,le DO à 2.., le DO de l'octave 0 à 37.

Une autre possibilité est de prendre comme repère le DO de l'octave 0, et de numéroter de manière algébrique.

Nous pouvons même asseoir notre formule sur un repère différent du LA international. Les programmes 1.2 et 1.4 utilisent le DO central.

Si vous observez toutes les règles le résultat doit être le même, ces autres systèmes de numérotation n'ayant pour seul but que de s'adapter à vos habitudes. Si nous entrons une grande quantité d'informations au clavier, il est généralement plus facile d'écrire la note plus le numéro de l'octave que de composer un nombre abstrait représentant la hauteur. DO0 est plus parlant que 478. De plus, une note exprimée en note+octave peut être analysée plus facilement, dans un programme, qu'un nombre. Nous examinerons ce point plus amplement au Chapitre 9.

Tout cela pour vous montrer qu'avec le couple micro-ordinateur / synthétiseur tout peut être envisagé. Nous adhérons à la numérotation utilisée dans le guide d'utilisation, mais nous énumérons ci-dessous quelques formules, pour ceux qui veulent utiliser une numérotation différente. Remplacez la formule de la ligne 170 (du dernier programme) par l'une de ces formules.

Si vous tenez à utiliser un incrément positif des octaves tel que vous le montre la figure 2.4, vous pouvez employer:

```
freq = 15.43375 *  2^(oct + note / 12)
```

en prenant pour base l'octave le plus bas, et ainsi vous n'aurez aucun compte négatif à faire. 1 = DO , etc...

```
freq = 30.8675 *   2^(note/12)
```

Vous noterez que nous n'avons plus besoin du paramètre oct **dans la dernière formule.** L'entrée des notes est réduite à la plus simple expression, mais semble, par la suite, plus difficile à exploiter qu'un couple note+octave. Du point de vue musical il vaut mieux garder la notation d'octave. Ces différentes numérotations ne sont citées que pour satisfaire votre curiosité d'esprit.

Construction microtonale des gammes

En partant de la même formule nous pouvons générer des gammes microtonales. Il suffit de remplacer le diviseur de gamme, 12, par le nombre de votre choix. C'est aussi simple que cela et votre AMSTRAD va vous en donner une démonstration:

```
100 REM PROGRAM 4.3
110 REM Gammes microtonales
120 :
130 oct=0
140 WHILE -1
150 INPUT"Combien d'intervalles";intv
160 PRINT
170 FOR note=0 TO intv
180 freq=261.262*2^(oct+note/intv)
190 hauteur=ROUND(125000/freq)
200 PRINT"Freq =";freq,"Hauteur =";
    hauteur
210 SOUND 1,hauteur
220 NEXT
230 WEND
```

Vous pouvez également entrer des valeurs inférieures à 12, ce qui donnera des gammes macrotonales. Nous trouvons des octaves de cinq ou sept intervalles dans la musique en provenance du Siam, de l'Arabie ou de Java.

Notez que la fréquence de base est celle du DO moyen. Si nous utilisions 440, notre gamme partirait du LA international; de toutes manières les résultats ne sont pas différents. Nous pouvons également changer d'octave si nécessaire.

Essayez le programme avec des intervalles de 12 à 20: la séquence ressemble à une suite de demi-tons, à moins que votre oreille ne soit très musicale. Le test réel démarrera quand vous essaierez de produire des accords et des mélodies. Il faut s'arrêter à ce niveau, sauf à envisager la publication d'un second tome. Espérant vous avoir mis en appétit, nous vous laissons à vos expériences...

L'avenir de la musique

Nous avons vu qu'avec un synthétiseur couplé à un ordinateur nous pouvons produire tous les sons immaginables. L'immense majorité de nos exemples portera sur la génération de musique conventionnelle, normale, ou du moins ce que la majorité des gens identifie comme tel. Après tout, le succès rencontré par cette forme de musique a prouvé que le choix était bon.

De nombreuses personnes, musiciens y compris, ne sont pas au courant qu'il existe d'autres formes de gammes. Nous avons détaillé cette étude des gammes car le domaine musical est en train d'évoluer, ainsi que le goût pour d'autres sonorités. La gamme microtonale en particulier commence à se vulgariser.

L'AMSTRAD nous permet de suivre cette évolution, nous pouvons expérimenter et produire de la musique sortant des limites habituelles. Le reste dépend de vous, de votre imagination et de votre ingéniosité.

5

ENV et ENT : les enveloppes de volume et de tonalité

A première vue, ces deux commandes semblent complexes. Il est vrai qu'elles peuvent admettre seize paramètres et être utilisées avec la commande SOUND. Néanmoins, en découpant le problème section par section nous allons trouver la question beaucoup plus facile. Nous allons nous apercevoir également que les commandes ENV et ENT peuvent prendre deux formes: la forme logicielle et la forme matérielle. Nous examinerons ENV en premier lieu.

ENV : l'enveloppe de volume

Le but de l'enveloppe de volume est d'altérer le volume d'une note pendant sa production. Nous avons eu une brève démonstration de la manière dont le son en est affecté, au chapitre 1, en expérimentant le Programme 1.4. Le volume initial sur lequel l'enveloppe de volume travaille est déterminé par la valeur V de la commande SOUND.

Le volume d'un son peut varier énormément durant sa production. Il peut y avoir six, sept niveaux différents, voire plus. La plupart des synthétiseurs du commerce offrent le contrôle sur quatre niveaux seulement, appelés: Attaque, Déclin, Soutien, Relâchement. Ces niveaux sont souvent désignés sous le sigle ADSR qui vient des termes anglais : "Attack, Decay, Sustain,

Release". Le fait qu'ADSR ne nous donne que quatre paramètres n'est pas une restriction cruelle, et suffit à vrai dire pour la plupart des sons et des instruments.

La commande ENV travaille différement et contrôle bien mieux l'amplitude, mais pour créer des tonalités d'instrument il est plus facile de se reporter aux notions ADSR.

ADSR : attaque, déclin, soutien, relâchement

Quoique ADSR soit très utilisé pour décrire les caractéristiques d'un instrument, le meilleur exemple pour expliciter les différentes phases d'un son est celui d'une voiture circulant sur une ligne droite et venant à notre rencontre. Nous l'entendons d'abord d'une manière très discrète, puis le son s'amplifie au fur et à mesure qu'elle se rapproche de nous. Quand elle se trouve à côté de nous le son est au maximum, et dès son dépassement le son faiblit immédiatement.Si le conducteur s'arrête un peu plus loin pour demander son chemin, le volume du son produit par son moteur restera constant quelques instants. Si nous dessinons une courbe représentant le volume sonore en fonction du temps, elle ressemblera à celle de la figure 5.1.

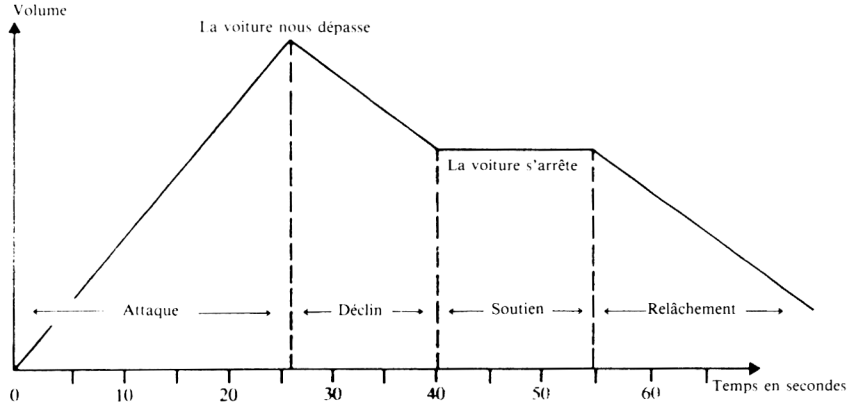


Figure 5.1. Le principe de l'ADSR.

L'exemple est un peu grossier et surtout se déroule sur une grande période de temps. Mais les variations de volume sont exactement les mêmes quand un instrument produit une note. Le son d'un instrument ne dure généralement pas plus d'une à deux secondes et souvent moins.

Pour plus de commodité, le profil du volume est divisé en différentes phases appelées: attaque, déclin, soutien et relâchement. Quelques sons ont plus de quatre phases et nous pouvons très bien utiliser la commande ENV pour produire des profils ou des enveloppes complexes. Nous allons d'abord examiner individuellement les quatre phases ADSR.

La phase d'attaque

Cette phase désigne le laps de temps qui s'écoule entre l'émission d'un son et le moment où il atteint un niveau particulier. En général ce niveau particulier est le volume maximal. Dans notre exemple de la voiture, la phase d'attaque était très longue et progressive. Les instruments à cordes ont une attaque très lente, de l'ordre du 1/10 de seconde. A l'opposé, nous trouvons les instruments de percussion (piano, batterie) et les instruments à cordes pincées comme la guitare, dont le volume atteint son maximum presque immédiatement. Leur durée d'attaque est d'environ 1/100 de seconde. Les cuivres et les instruments à vent comme les clarinettes ont un temps d'attaque moyen.

La phase de déclin

Cette phase se produit immédiatement après la phase d'attaque. C'est le laps de temps que met le volume à passer du maximum à un seuil inférieur appelé seuil de soutien. Dans cette phase, ainsi que le suggère son nom, le niveau sonore diminue légèrement. Pour quelques instruments, le niveau a tendance à rester identique, mais ceci ne se produit que si le niveau de soutien est identique au niveau atteint à la fin de la phase d'attaque. Dans ce

cas, la phase de déclin est réduite à zéro et l'enveloppe de volume passe directement de la phase d'attaque à la phase de soutien.

Il se peut aussi que la phase de déclin soit plus longue que la phase d'attaque. Certains instruments, comme la trompette, entrent en oscillation et les vibrations ne peuvent être arrêtées d'une manière brutale. Si nous revenons à notre exemple, nous pouvons trouver une certaine analogie dans le fait que la voiture ne peut pas s'arrêter d'une manière brutale sans ralentir. Les sons issus d'un instrument demandent, au minimum,quelques millisecondes pour être étouffés. Ceci explique pourquoi les enregistrements de musique passés à l'envers semblent si étranges: les sons meurent trop rapidement. Ce phénomène peut être simulé très facilement par des moyens électroniques. L'AMSTRAD le fait très aisément par la commande ENV, comme nous l'avons vu dans l'exemple 1.4.

Avec la phase de déclin nous sommes pratiquement arrivés au stade final d'un volume sonore. Beaucoup de gens pensent qu'après le stade de déclin on passe directement au stade de relâchement. Mais il apparaît une autre nuance, que nous allons examiner rapidement.

La phase de soutien

Après son déclin, le son entre dans une phase de soutien. Comme ce mot l'indique, le son est alors maintenu à un niveau sonore constant. Cette phase n'est pas exprimée en secondes ou en fractions de seconde, comme le sont les autres phases - ce qui amène souvent à des confusions. Il faut raisonner en niveau sonore. Pour les joueurs d'orgue électronique, par exemple, la phase de soutien est le volume sonore auquel la note est maintenue, aussi longtemps que la touche est enfoncée.

Cette explication peut sembler confuse, ceci est dû à la flexibilité de l'enveloppe ADSR. Le programme suivant et les explications qui

l'accompagnent vous permettront de voir et d'entendre de quoi je veux parler. Surtout n'abandonnez pas.

La phase de relâchement

Après avoir traversé les trois phases, le son atteint la phase où il devra terminer son cheminement. Il peut mourir lentement ou être coupé de manière plus ou moins brutale. Cette phase est appelée: phase de relâchement. Le vibraphone et plusieurs autres instruments de percussion ont une phase de relâchement très longue. Les instruments à cordes ont une période un peu moins longue. Cette phase est mesurée en temps, comme la phase d'attaque ou de déclin, et détermine à quelle vitesse la note va mourir.

La remarque que j'ai faite précédemment, indiquant que le temps de déclin était souvent plus long que le temps d'attaque, s'applique également à cette phase. De nombreuses personnes, à tort, se réfèrent à la phase de relâchement comme à la phase de déclin. ADSR est juste une convention pour séparer l'enveloppe de volume en plusieurs parties facilement exploitables. Il est peu probable que des confusions puissent surgir à partir du moment où le contexte a été clairement défini. Sur un synthétiseur, la phase de relâchement démarre au moment où vous relâchez la touche du clavier. Si vous appuyez immédiatement sur une nouvelle touche, la phase de relâchement sera couverte par la phase d'attaque de la note suivante.

Une enveloppe ADSR complète:la mise en commun

L'enveloppe de volume de la plupart des instruments et de nombreux sons peut être définie avec précision avec l'ADSR. Vous avez sûrement réalisé que tous les sons n'ont pas besoin des quatre phases. Un orgue électronique, par exemple, a une phase d'attaque, de soutien et de relâchement, mais

pas de phase de soutien. Les bois ont une attaque et une phase de relâchement extrêmement rapides, une corde de guitare ou de banjo ont une attaque et un relâchement plus lents. Quelques exemples en figure 5.2. vous éclaireront.

Vous avez compris, ou vous allez comprendre après avoir passé ce programme, qu'il n'est pas toujours possible de déterminer où s'arrête une phase pour laisser la place à une autre. Souvent la phase de déclin passe directement à la phase de relâchement. Pour une note isolée la différence n'est pas importante, mais si elle est suivie immédiatement d'une autre note et que son effet de déclin est produit par la phase de relâchement, l'effet peut être réduit.

Vous pouvez programmer des sons sans phase de déclin ou de soutien. Quelques synthétiseurs ont un générateur d'enveloppe très réduit, du AR, qui permet seulement de programmer la phase d'attaque et de relâchement.

Nous allons voir comment la commande ENV peut être utilisée pour créer l'enveloppe de type ADSR, ou même d'autres types d'enveloppes.

LES PARAMETRES ENV

Nous nous concentrerons sur la définition logicielle en premier. Le guide d'utilisation nous donne la liste des paramètres de la manière suivante:

ENV N, P1, Q1, R1, P2, Q2, R2, P3, Q3, R3, P4, Q4, R4, P5, Q5, R5

et N est le numéro de l'enveloppe. Les lettres **P,Q,R,** ne sont pas très mnémoniques, mais chaque groupe de **P, Q, R,** est structuré de manière identique et nous n'avons que trois lettres à retenir.

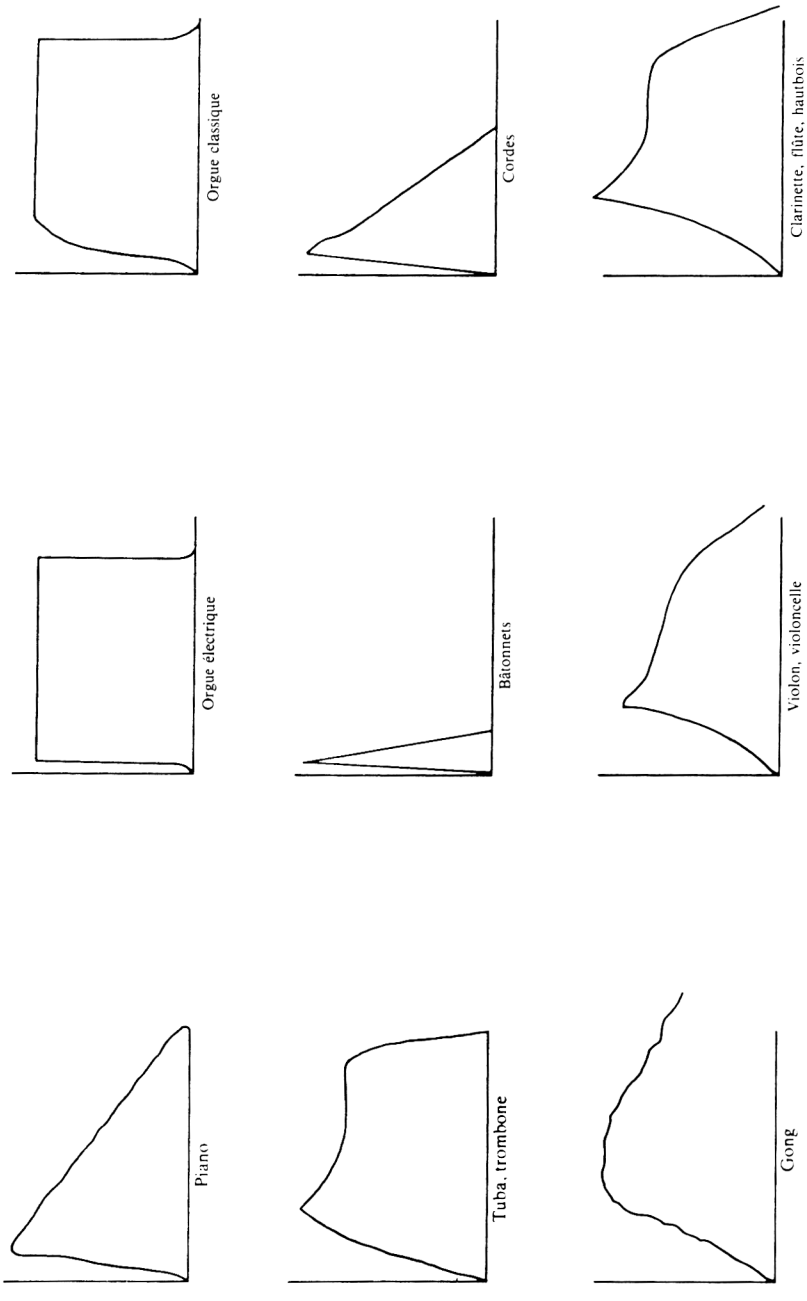


Figure 5.2. Exemples d'enveloppes.

Nous pouvons identifier chaque groupe de P, Q, R, comme étant une partie, ou une section de l'enveloppe. Les paramètres et leurs valeurs limites sont résumés en figure 5.3.

Paramètre	Nom	Spectre
N	Numéro d'enveloppe	1 à 15
Logiciel	P, Q, R d'ENV N	
P Q R	Nombre d'intervalles Taille de l'intervalle Temps de la pause	0 à 127 – 128 à 127 0 à 255
Matériel	= HE, EP d'ENV N	
= HE EP	Enveloppe matérielle Période de l'enveloppe	0 à 15 – 32.768 à 65.535

Figure 5.3. Les paramètres d'enveloppe et leur échelle de valeur.

Après avoir défini une enveloppe, l'ordinateur garde en mémoire les paramètres. Une commande ENV est remise à zéro en entrant ENV sans aucun paramètre; par exemple:

```
ENV 1
```

P,Q,R: une section de l'enveloppe de volume

Chaque section de P, Q, R se suffit à elle-même, et nous pouvons avoir jusqu'à cinq sections. Elles opèrent toutes de la même manière. La section doit être définie complètement, mais leur nombre est optionnel.

P spécifie le nombre de pas de chaque section, Q l'importance de la variation de volume dans chaque section, et R la durée de chaque pas en 1/100 de seconde.

Si nous utilisons une enveloppe de volume avec la commande SOUND, le volume peut être

modulé sur seize niveaux au maximum, de 0 à 15. La figure 5.4 nous montre comment une commande ENV peut être définie pour produire une enveloppe du type ADSR. Le paramètre V de la commande SOUND définit le volume initial. Si vous utilisez ENV, il est judicieux de mettre V à 0. Ceci permet un contrôle total du volume par ENV, et s'il arrive des sons bizarres au moins V ne pourra être mis en cause. Vous pouvez voir sur la figure de quelle manière les paramètres affectent le volume.

Valeurs spéciales de P,Q,R

Valeurs spéciales de P,Q,R : de même que pour la commande SOUND, certaines valeurs mettent la commande ENV dans des conditions spéciales. Si P=0, l'amplitude résultante est la valeur absolue de Q et durera le temps spécifié par R.

Il se produit un recouvrement du niveau du volume si Q prend des valeurs extérieures aux limites (16 produira un niveau 0, 17 un niveau 1, etc.). Ceci peut être utilisé pour créer une enveloppe mono-section (programme 1.4).

Si Q=0, le niveau sonore restera à l'état initial pour une durée de P x R (voir section 3, figure 5.4).

Si P et Q sont tous deux à 0, alors le niveau sonore sera 0 pendant la durée de R.

Une valeur R=0 est traitée comme si elle était égale à 256.

Si le paramètre D de la commande SOUND prend une valeur positive et supérieure à la durée de l'enveloppe, la durée restante aura le volume sonore qu'a atteint le son. Si le paramètre D est inférieur à la durée ENV, la durée excédentaire de l'enveloppe sera ignorée.

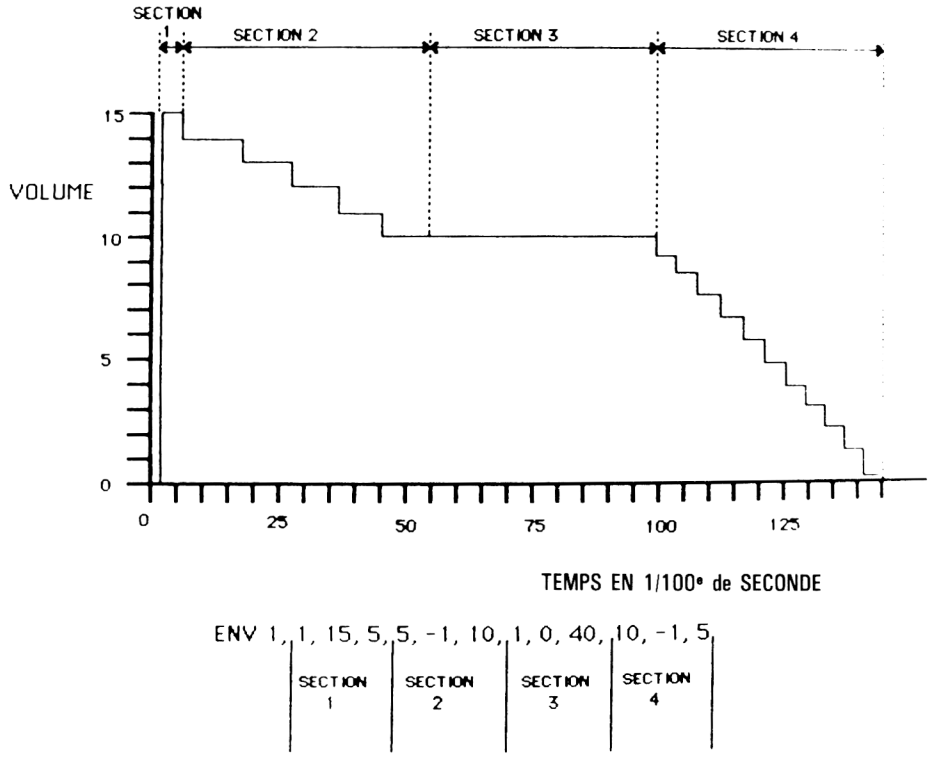


Figure 5.4. L'enveloppe de volume sous sa forme typique d'enveloppe ADSR.

UN GENERATEUR GRAPHIQUE DE L'ENVELOPPE

La meilleure façon d'exploiter les commandes ENT et ENV est d'écouter et de visualiser leurs effets. Le programme suivant vous permet de modifier les paramètres, d'écouter et de visualiser les différents effets sur ENV. Il est possible de faire la même chose avec l'enveloppe ENT, mais pour l'instant cette commande n'est pas activée.

```
1000 REM PROGRAM 5.1
1010 REM Générateur graphique d'ENV et
     ENT
1020 :
1030 MODE 1
1040 GOSUB 1270:REM Mise au point
```

```
1050 GOSUB 1600:REM Dessiner l'axe
1060 GOSUB 1750:REM Imprimer ENV,ENT
1070 :
1080 WHILE -1
1090 com$=INKEY$:IF com$="" GOTO 1090
1100 com=ASC(LOWER$(com$))
1110 IF com=13 OR com=32 THEN GOSUB 2550
     :GOTO 1090:REM Son et graphique
1120 IF com=100 THEN GOSUB 1940:GOTO 109
     0:REM Changer la durée de la note
1130 IF com=112 THEN GOSUB 1990:GOTO
     1090:REM Changer le hauteur
1140 IF com=121 THEN GOSUB 2040:GOTO
     1090:REM Changer Yscale dans
     graphique ENT
1150 IF com=119 THEN GOSUB 1600:GOTO
     1090:REM CLG & Dessiner Axe
1160 IF com=105 THEN ON vort GOSUB 1520,
     1550:REM Enveloppe Initiale
1170 IF com=99 THEN GOSUB 1880:REM
     Effacer ENV, ENT
1180 IF com=114 THEN LOCATE 4,3:etr=etr*
     -1:IF etr=-1 THEN PRINT "-";:GOTO
     1090 ELSE PRINT "-";:GOTO 1090:REM
     Interruption/reprise répétition ENT
1190 IF com=242 OR com=243 THEN GOSUB
     2100:GOTO 1220:REM Choisir un
     paramètre
1200 IF com>239 AND com<246 THEN GOSUB
     2170:REM Changer le paramètre
1210 IF com=9 THEN pn=1:GOSUB 1750:pn=2:
     vort=vort XOR 3:REM Echange ENV/ENT
     :pn=PEN noir
1220 GOSUB 1750:REM Imprimer ENV, ENT
1230 WEND
1240 END
1250 :
1260 REM Mise au point
1270 INK 0,12:INK 1,0:INK 2,26:INK 3,10
1280 BORDER 10
1290 DEFINT a-f,h-s
1300 DIM ev(15),et(15)
1310 etr=1:REM Répétition d'ENT
1320 cpos=1:REM Position du curseur
1330 dur=25
1340 hauteur=956
```

```
1350 vort=1:REM enVORenT:1=ENV 2=ENT
1360 pn=2:REM PEN = noir
1370 yscale=256/4095:ysca=1
1380 :
1390 WINDOW #1,1,40,1,7
1400 WINDOW SWAP 0,1
1410 PAPER 3:CLS
1420 PRINT "ENV 1";
1430 LOCATE 1,3:PRINT "ENT";etr;
1440 PEN 2
1450 LOCATE 10,6:PRINT"Y";SPC(15)"P";
1460 LOCATE 1,7:PRINT "W";SPC(4)"C";SPC
     (5)"I";SPC(7)"R";SPC(6)"D";
1470 PEN 1
1480 LOCATE 11,6:PRINT CHR$(22)+CHR$(1);
     "gamme 1:";ysca;"    hauteur:
     "hauteur;
1490 LOCATE 2,7:PRINT "ipe lear nitial
     epeat  ur :";dur;CHR$(22)+CHR$(0);
1500 ORIGIN 32,32,0,639,287,0
1510 :
1520 ev(1)=1:ev(2)=15:ev(3)=5:ev(4)=5:ev
     (5)=-1:ev(6)=10:ev(7)=1:ev(8)=0:ev
     (9)=50:ev(10)=10:ev(11)=-1:ev(12)=5
     :ev(13)=0:ev(14)=0:ev(15)=0
1530 REM 16 sec ENV:ev(1)=1:ev(2)=15:ev
     (3)=200:ev(4)=14:ev(5)=0:ev(6)=100:
     ev(7)=0:ev(8)=0:ev(9)=0:ev(10)=0:
     (11)=0:ev(12)=0:ev(13)=0:ev(14)=0:
     ev(15)=0
1540 RETURN
1550 et(1)=10:et(2)=32:et(3)=10:et(4)=40
     :et(5)=-16:et(6)=5:et(7)=0:et(8)=0:
     et(9)=0:et(10)=0:et(11)=0:et(12)=0:
     et(13)=0:et(14)=0:et(15)=0
1560 REM Vibrato:et(1)=1:et(2)=1:et(3)=5
     :et(4)=2:et(5)=-1:et(6)=5:et(7)=1:
     et(8)=1:et(9)=5:et(10)=0:et(11)=0:
     et(12)=0:et(13)=0:et(14)=0:et(15)=0
1570 RETURN
1580 :
1590 REM Dessiner l'axe
1600 CLG 0
1610 MOVE 0,0:DRAW 592,0,1
1620 MOVE 0,-16:DRAW 0,240
1630 TAG
1640 FOR n=15 TO 255 STEP 16
```

```
1650 MOVE -16,n:PRINT "_";
1660 NEXT n
1670 TAGOFF
1680 WINDOW SWAP 1,0
1690 LOCATE 3,25
1700 PAPER 0:PRINT  "Temps d'ENV:
     Temps d'ENT:";
1710 WINDOW SWAP 0,1
1720 RETURN
1730 :
1740 REM Imprimer ENV, ENT
1750 LOCATE 6,vort OR 1:REM 1 ou 3
1760 FOR n=1 TO 15
1770 IF vort=1 THEN e$=STR$(ev(n)) ELSE
     e$=STR$(et(n))
1780 IF VAL(e$)>=0 THEN e$=RIGHT$(e$,LEN
     (e$)-1)
1790 PRINT ",";
1800 IF cpos=n THEN PEN pn
1810 PRINT USING "&";e$;
1820 PEN 1
1830 NEXT n
1840 IF com=99 THEN PRINT SPC(35) ELSE
     PRINT SPC(4)
1850 RETURN
1860 :
1870 REM Effacer ENV ou ENT
1880 cpos=1
1890 IF vort=2 THEN 1910
1900 FOR n=1 TO 15:ev(n)=0:NEXT:RETURN
1910 FOR n=1 TO 15:et(n)=0:NEXT:RETURN
1920 :
1930 REM Changer la durée de la note
1940 LOCATE 32,7:PRINT SPC(4):LOCATE 32,
     7:INPUT dur
1950 LOCATE 32,7:PRINT dur;
1960 RETURN
1970 :
1980 REM Changer la hauteur
1990 LOCATE 32,6:PRINT SPC(5):LOCATE 32,
     6:INPUT hauteur
2000 LOCATE 32,6:PRINT hauteur;
2010 RETURN
2020 :
2030 REM Changer Yscale dans graph ENT
2040 LOCATE 19,6:PRINT SPC(4):LOCATE 19,
     6:INPUT ysca
```

```
2050 LOCATE 19,6:PRINT ysca;" ";
2060 yscale=256/4095*ysca
2070 RETURN
2080 :
2090 REM Choisir un paramètre
2100 IF com=242 THEN cpos=cpos-1
2110 IF com=243 THEN cpos=cpos+1
2120 IF cpos>15 THEN cpos=1
2130 IF cpos<1 THEN cpos=15
2140 RETURN
2150 :
2160 REM Changer p,q,r et t,v,w
2170 IF com=240 THEN inc=1
2180 IF com=241 THEN inc=-1
2190 IF com=244 THEN inc=10
2200 IF com=245 THEN inc=-10
2210 :
2220 IF vort=2 THEN 2250
2230 ev(cpos)=ev(cpos)+inc
2240 ON ((cpos+2) MOD 3)+1 GOTO 2290,
     2330,2370
2250 et(cpos)=et(cpos)+inc
2260 ON ((cpos+2) MOD 3)+1 GOTO 2420,
     2460,2500
2270 :
2280 REM p
2290 IF ev(cpos)<0 THEN ev(cpos)=127
2300 IF ev(cpos)>127 THEN ev(cpos)=0
2310 RETURN
2320 REM q
2330 IF ev(cpos)<-128 THEN ev(cpos)=127
2340 IF ev(cpos)>127 THEN ev(cpos)=-128
2350 RETURN
2360 REM r
2370 IF ev(cpos)<0 THEN ev(cpos)=255
2380 IF ev(cpos)>255 THEN ev(cpos)=0
2390 RETURN
2400 :
2410 REM t
2420 IF et(cpos)<0 THEN et(cpos)=239
2430 IF et(cpos)>239 THEN et(cpos)=0
2440 RETURN
2450 REM v
2460 IF et(cpos)<-128 THEN et(cpos)=127
2470 IF et(cpos)>127 THEN et(cpos)=-128
2480 RETURN
```

```
2490 REM w
2500 IF et(cpos)<0 THEN et(cpos)=255
2510 IF et(cpos)>255 THEN et(cpos)=0
2520 RETURN
2530 :
2540 REM Routine son et graphique
2550 GOSUB 2720:REM Prendre dernière
     section ENV
2560 IF vlast=0 THEN LOCATE 1,5:PRINT "
     Aucune section dans cette ENV.":FOR
     d=1 TO 3000:NEXT:LOCATE 1,5:PRINT
     SPC(34):RETURN
2570 ENV 1,ev(1),ev(2),ev(3),ev(4),ev(5)
     ,ev(6),ev(7),ev(8),ev(9),ev(10),ev
     (11),ev(12),ev(13),ev(14),ev(15)
2580 POKE(&B61A),vlast\3+1+128:REM \ est
     au-dessus de CTRL
2590 :
2600 GOSUB 2790:REM Prendre dernière
     section ENT
2610 IF tlast=0 THEN POKE(&B70A),0:GOTO
     2660:REM Sauter Définition ENT
2620 :
2630 ENT etr,et(1),et(2),et(3),et(4),et
     (5),et(6),et(7),et(8),et(9),et(10),
     et(11),et(12),et(13),et(14),et(15)
2640 IF etr=1 THEN POKE(&B70A),tlast\3+1
     ELSE POKE(&B70A),tlast\3+1+128:REM
     \ est au-dessus de CTRL
2650 :
2660 GOSUB 3560:REM Jouer note ou air
2670 GOSUB 2860:REM Temps d'ENV & ENT
2680 ON vort GOSUB 3100,3290:REM Impres
     sion du graphique d'ENV ou ENT
2690 RETURN
2700 :
2710 REM Trouver dernière section d'ENV
2720 vlast=0
2730 FOR j=13 TO 1 STEP -3
2740 IF NOT(ev(j)=0 AND ev(j+1)=0 AND ev
     (j+2)=0) THEN vlast=j:j=1
2750 NEXT j
2760 RETURN
2770 :
2780 REM Trouver dernière section d'ENT
2790 tlast=0
```

```
2800 FOR j=13 TO 1 STEP -3
2810 IF NOT(et(j)=0 AND et(j+1)=0 AND et
     (j+2)=0) THEN tlast=j:j=1
2820 NEXT j
2830 RETURN
2840 :
2850 REM Trouver le temps d'ENV
2860 vtim=0
2870 FOR n=1 TO vlast STEP 3
2880 pt=ev(n+2):IF pt=0 THEN pt=256:REM
     Si temps pause =0 alors comme 256
2890 IF ev(n)=0 THEN vtim=vtim+pt:GOTO
     2910:REM Nombre d'interval=0
2900 vtim=vtim+ev(n)*pt:REM Nombre d'
     interval*temps de pause
2910 NEXT n
2920 :
2930 REM Trouver le temps d'ENT
2940 ttim=0
2950 FOR n=1 TO tlast STEP 3
2960 pt=et(n+2):IF pt=0 THEN pt=256:REM
     Si temps pause =0 alors comme 256
2970 IF et(n)=0 THEN ttim=ttim+pt:GOTO
     2990:REM Interval=0 comme 1
2980 ttim=ttim+et(n)*pt:REM Nombre d'
     interval*temps de pause
2990 NEXT
3000 :
3010 REM Imprimer les temps
3020 WINDOW SWAP 1,0
3030 LOCATE 12,25:PRINT SPC(8)
3040 PAPER 0:LOCATE 12,25:PRINT vtim/100
     ;
3050 LOCATE 32,25:PRINT SPC(8):LOCATE 32
     ,25:PRINT ttim/100;
3060 WINDOW SWAP 0,1
3070 RETURN
3080 :
3090 REM Dessiner ENV
3100 xscale=ROUND(592/vtim,2)
3110 MOVE 0,0
3120 FOR n=1 TO vlast STEP 3
3130 gxpos=ev(n+2):IF gxpos=0 THEN gxpos
     =256:REM Si temps pause =0 alors
     comme 256
3140 gxpos=gxpos*xscale
```

```
3150 gypos=(ev(n+1) MOD 16)*16:REM Chaq
     interval=16 unités graphiques
3160 IF gypos<0 THEN gypos=gypos+256:REM
     Effet de repli
3170 IF ev(n)=0 THEN DRAW XPOS,gypos,2:
     DRAWR gxpos,0:GOTO 3250:REM Si nomb
     d'interval=0 alors Amp=taille abso
     lue de l'intervalle
3180 :
3190 REM Nombre d'intervalles
3200 FOR g=1 TO ev(n)
3210 IF YPOS+gypos>=256 THEN DRAW XPOS,
     YPOS+gypos-256,2:DRAWR gxpos,0:GOTO
     3230:REM Si répétés, intervalles
     sortiront graphique de l'écran
3220 DRAWR 0,gypos,2:DRAWR gxpos,0
3230 NEXT g
3240 :
3250 NEXT n
3260 RETURN
3270 :
3280 REM Dessiner ENT
3290 IF tlast=0 THEN LOCATE 1,5:PRINT "E
     NT vide":FOR d=1 TO 3000:NEXT:
     LOCATE 1,5:PRINT SPC(18):RETURN
3300 :
3310 MOVE 0,INT(hauteur/4095*256)
3320 DRAWR 592,0,1:REM Axe X d' hauteur
3330 MOVE 0,INT(hauteur/4095*256)
3340 xscale=ROUND(592/ttim,2)
3350 gypos=hauteur
3360 :
3370 FOR n=1 TO tlast STEP 3
3380 gxpos=et(n+2):IF gxpos=0 THEN gxpos
     =256:REM Si temps pause=0 alors
     comme 256
3390 gxpos=gxpos*xscale
3400 gystep=et(n+1)
3410 :
3420 REM Nombre d'intervalles
3430 IF et(n)=0 THEN scount=1 ELSE
     scount=et(n):REM 0 interval= 1
3440 FOR g=1 TO scount
3450 IF gypos+gystep>4095 THEN gypos=
     gypos+gystep-4096:DRAW XPOS,gypos*
     yscale,2:GOTO 3490
```

```
3460 IF gypos+gystep<0 THEN gypos=4096+
     gypos+gystep:DRAW XPOS,gypos*yscale
     ,2:GOTO 3490
3470 gypos=gypos+gystep
3480 DRAWR 0,gystep*yscale,2
3490 DRAWR gxpos,0
3500 NEXT g
3510 :
3520 NEXT n
3530 RETURN
3540 :
3550 REM Jouer note ou air
3560 IF com=13 THEN SOUND 129,hauteur,0,
     0,1,1:RETURN:REM Jouer note
3570 :
3580 REM Jouer air
3590 SOUND 129,0,1,0:REM Vider le canal
3600 RESTORE
3610 FOR n=1 TO 7:READ note
3620 SOUND 1,note,dur,0,1,1
3630 NEXT n
3640 DATA 478,478,536,478,402,478,478
3650 RETURN
```

Commentaires

Une description des routines affectant l'enveloppe de tonalité sera donnée en temps utile. L'initialisation des variables, des couleurs de l'écran, etc., est faite en ligne 1270. La variable **array, ev,** contient les paramètres de l'enveloppe de volume, et la variable **et** les paramètres de tonalité. **yscale** et **ysca** sont utilisés pour changer l'axe Y sur l'ENT graphique. CHR$(22) en lignes 1480 et 1490 positionne l'inversion des couleurs pour que les lettres noires ne se greffent pas sur les initiales blanches (voir le guide d'utilisation Chapitre 5 page 2). Une fenêtre graphique est définie en ligne 1500 avec une nouvelle origine, correspondant à l'intersection des axes X et Y. En 1600 on trouve la routine qui dessine les axes du graphique. Celle qui démarre en ligne 1750 affiche les définitions de l'enveloppe

sur l'écran. vort OR 1 est un raccourci pour retourner à une ou trois valeurs. Le Basic de l'AMSTRAD a une manière bien à lui d'afficher, très utile pour les tableaux, mais qui ne nous convient pas dans cette application. C'est pourquoi je suis obligé de mettre les paramètres de forme alphanumérique en lignes 1770 et 1780. Ainsi les espaces sur les nombres positifs sont supprimés. Si la position du curseur est la même que le nombre de paramètres, le paramètre sera imprimé en blanc. En ligne 1480, des espaces sont affichés pour nettoyer l'écran.

Après toutes ces instructions, le contrôle est passé à la ligne 1090 qui attend la pression d'une touche de commande; cette commande sera convertie en valeur numérique puis stockée dans la variable **com.** Nous allons examiner ces routines dans l'ordre où elles apparaissent dans le programme.

La routine de la ligne 1880, place le curseur au départ de la définition de l'enveloppe et remet tous les paramètres à zéro. Les trois routines qui suivent changent la durée, la valeur de tonalité et le paramètre **yscale.** Elles sont similaires et très simples.

Les lignes 2170 et 2200 déterminent la valeur qui va altérer le paramètre. La ligne 2220 nous dirige vers ENT, si cette option a été choisie. En 2230 et 2250 nous ajouterons l'incrément et nous ferons un saut pour vérifier que le paramètre n'est pas en dehors des limites. Là aussi nous retrouvons une boucle. La routine de la ligne 2540 (son et graphique) est celle qui fait le plus gros du travail. Pour bien comprendre son architecture, il nous faut connaître les enveloppes reconnues par l'ordinateur.

COMBIEN DE SECTIONS A UNE ENVELOPPE?

Le fait d'entrer une définition d'enveloppe dans le micro détermine le nombre de sections. Par exemple avec:

```
ENV 1,1,15,10
```

étant donné qu'il n'y a qu'un jeu de P, Q, R, il n'y a qu'une seule section. avec:

```
ENV 1,1,15,10,1,-5,20
```

il y a deux sections, etc.

Si nous entrons:

```
ENV 1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

nous n'avons pas fait une remise à blanc de la commande ENV, mais nous avons défini une enveloppe à cinq sections. Le paramètre R=0 agit comme s'il avait la valeur 256 et la durée est égale à 5x256, le volume par contre est de 0.

Techniquement, pour programmer une enveloppe avec moins de cinq sections nous devons passer à la forme raccourcie et non mettre à zéro les sections non utilisées (voir lignes 2570 et 2630). Comme pour toutes les informations, le micro assure le stockage dans les zones-mémoire correspondant aux différentes commandes. Pour ENV1 l'adresse-mémoire qui spécifie le nombre de sections est: &B61a et pour ENT: &B70a. Nous parlerons plus longuement de cette zone à la fin de ce chapitre.

Dans le programme, il nous faut connaître le nombre de sections programmées. Il s'assure que tout groupe de paramètres à 0 soit considéré comme non utilisé. La routine (son et graphique) effectue un appel à la routine 2720 pour déterminer la dernière section de ENV. Cette routine ainsi que celle en ligne 2790, pour l'enveloppe de tonalité, scrute tous les paramètres à la recherche des groupes P,Q,R ne comportant pas trois zéros consécutifs. S'il ne trouve aucune section avec des paramètres différents de zéro, il n'y a pas de section du tout. La ligne 2740 semble un peu compliquée, mais elle ne fait que mettre en place cette logique de scrutation. Donc, dans toutes les

enveloppes entrées au micro-ordinateur, vous pouvez omettre les sections ne comportant pas de zéro.

Si une commande ENV à blanc a été définie, un message est imprimé et la routine continue. Dans l'autre cas, le nombre de sections est écrit par l'instruction POKE dans l'adresse mémoire correspondante. Les lignes 2600 à 2640 font exactement la même chose pour l'enveloppe de tonalité. Regardez la ligne 2640. Si une répétition a été placée, l'ordinateur met en mémoire l'information en ajoutant 128, le bit 7, à la valeur de l'enveloppe.

Remarquez le caractère "ç" en ligne 2580 et 2640. Ce symbole nous permet de ne récupérer que la partie entière du reste d'une division. La ligne 2660 fait appel à une routine pour jouer une note ou une tonalité. Celle-ci remplit la file d'attente et nous redonne la main pour dessiner le graphique. Le calcul de la longueur de l'enveloppe est fait par la routine appelée en 2670. Le temps est imprimé au bas du graphique. La routine du dessin de la fonction est appelée en ligne 2680. Cette routine est un peu compliquée, elle doit prendre en compte toutes les variantes d'une enveloppe.

Le graphique ENV calcule l'échelle X en 3100, en tenant compte de la longueur de l'enveloppe. La ligne 3120 scrute toutes les sections des enveloppes et **gxpos** maintient la position du curseur sur l'axe des X. **gypos** fait de même pour Y, mais est ajusté en ligne 3160 pour prendre en considération l'effet de recouvrement. La ligne 3170 dessine le niveau du volume, si le nombre d'intervalles est 0.

La boucle entre les lignes 3200 et 3230 balaye les différents intervalles du niveau du volume. La plupart des lignes sont dessinées en position relative, par rapport aux derniers points avec DRAWR, mais s'il y a recouvrement 3210 dessine le niveau de volume correct.

Dans le cas où l'on a choisi et appelé ENT, la routine en 3290 dessine l'enveloppe de tonalité. Si l'enveloppe est vide, nous le saurons dès la première ligne. La ligne 3310 retrace proportionnellement la fonction sur l'axe des X. **xscale** est déterminé de la même manière que dans le graphique ENV. **gypos** est égal à la hauteur. Comme pour ENV, le programme scrute chaque section de l'enveloppe. **gxpos** est égal au temps de pause et **gystep** est fonction de la grandeur de l'intervalle. Le programme boucle sur le nombre d'intervalles et incrémente **gypos** et **gystep.** 3450 et 3460 s'occupent de l'effet de recouvrement. Elles utilisent **yscale** défini en 1370 et qui peut être modifié en appuyant la touche Y.

gystep repère la hauteur exacte de la note. Nous n'avons que 256 unités graphiques à notre disposition, or nous essayons de travailler sur une résolution de 4096. Nous ne pouvons donc travailler en termes d'unités graphiques comme nous l'avions fait avec ENV. Si aucun effet de recouvrement ne se produit, 3470 et 3480 peuvent dessiner normalement les changements de hauteur.

La dernière routine est chargée de jouer la note ou l'accord. L'accord est composé de sept notes. Si la durée de la note est très longue, la file d'attente va être remplie très rapidement et le programme se bloquera. Si vous voulez expérimenter des notes très longues, réduisez l'accord à cinq notes et faites l'essai suivant:

```
3640 DATA 213,190,239,478,319
```

Utilisation du programme

Le programme imprime les cinq sections de ENV1 sur le haut de l'écran. En dessous ENT1 est en attente d'activation; nous nous occuperons de l'activer plus loin dans ce chapitre. L'axe Y est divisé en seize niveaux de volumes, l'axe X représente le temps. Comme la durée d'une enveloppe peut varier entre 1/100 de

seconde et plusieurs minutes, l'échelle choisie est celle qui permet de remplir l'écran au mieux. Le temps des enveloppes est imprimé sur le bas, en secondes.

Si vous appuyez sur la touche RETOUR, l'enveloppe est jouée et son dessin apparaît sur l'écran. En pressant P vous pouvez changer la hauteur de la note, sachant que cette valeur doit être comprise entre 0 et 4095. Après avoir frappé la nouvelle valeur, appuyez sur RETOUR. Si vous pressez sur la barre d'espacement, un son très faible est émis, il vous permet de jauger l'effet de l'enveloppe.

Les différentes représentations graphiques se superposent sur l'écran, mais la touche W peut vous nettoyer l'écran si nécessaire.

La touche C remet à zéro tous les paramètres de l'enveloppe et positionne le curseur sur le premier paramètre pour pouvoir entrer une nouvelle enveloppe. I réinitialise l'enveloppe avec ses anciens paramètres de la routine de mise au point. Y et R sont des touches spécifiques aux enveloppes de tonalité et de volume et seront explicitées ultérieurement.

Les paramètres de l'enveloppe sont changés avec les touches de positionnement du curseur. La flèche gauche et la flèche droite affichent un paramètre après l'autre dans un blanc plus intense. Les flèches haute et basse changent la valeur des paramètres en incrémentant de 1 à chaque fois. Si en même temps vous appuyez sur la touche des majuscules l'incrément passe de 1 à 10. L'autre touche de contrôle est TAB qui passe alternativement de ENV à ENT.

ETUDE DES ENVELOPPES

Une approche simple de l'étude des enveloppes peut être la production d'une section ADSR avec la commande ENV. Vous pouvez en voir le résultat sur l'écran en exécutant l'enveloppe

initiale. C'est un bon moyen de vous familiariser avec les paramètres.

Vous avez remarqué le recouvrement quand les valeurs sortent des limites, et le graphique nous le montre clairement. Ce recouvrement peut être utilisé pour produire des enveloppes complexes, mais également pour créer des enveloppes à une seule section, très pratiques. Le programme 1.4 utilise une enveloppe à une seule section très percutante.

```
ENV 1,16,15,14
```

Entrez cela dans le programme et changez le paramètre Q de 1 à 16. Ceci est juste une démonstration pour voir la complexité des enveloppes que l'on peut créer.

A cause du recouvrement Q=1 correspondra à Q=17, Q=2 à Q=18, etc.

A partir de l'exemple en figure 5.2, essayez de produire des enveloppes ENV. Dans la suite nous allons voir comment ENV peut être utilisé pour produire des effets musicaux. Notez les enveloppes intéressantes que vous allez inventer.

LA FORME MATERIELLE D'ENV

Vous venez à peine de comprendre l'enveloppe logicielle, voici qu'apparaît l'enveloppe matérielle. Le guide d'utilisation ne fait que signaler son existence, sans plus. Vous pouvez ignorer tout cela, sans subir de perte pour vos programmes musicaux, mais vous verrez qu'elle peut vous servir à produire des effets qu'il est difficile de produire avec la forme logicielle.

Une enveloppe matérielle constitue une alternative avec les paramètres P,Q,R de la section logiciel et consiste uniquement en deux paramètres au lieu de trois. Ils prennent la forme suivante:

=HE,EP

où HE est le numéro de l'enveloppe et EP la période de l'enveloppe. HE peut prendre les valeurs de 0 à 15. EP détermine la longueur des intervalles en unités de 128 microsecondes et prend les valeurs comprises entre -32.768 et 65.535.

Les enveloppes matérielles ont des caractéristiques prédéterminées et vous pouvez mélanger des sections matérielles et logicielles dans une même définition d'enveloppe. Cependant les sections matérielles n'ayant pas un temps de pause associé, elles doivent être suivies par une section logicielle qui précise le temps de pause. Par exemple:

```
ENV 1,=he,ep,5,0,100
```

va produire une pause de 5 secondes.

Si deux ou plusieurs sections matérielles consécutives sont définies, le contrôle va être fait dans la dernière section. Si une commande ENV se termine avec une enveloppe matérielle, la durée par défaut sera de 2 secondes.

Les enveloppes principales vont de 8 à 15. Les enveloppes de 0 à 3 sont identiques à l'enveloppe 9, et 4 à 7 sont identiques à l'enveloppe 15. Voici un petit programme qui vous fera une démonstration:

```
100 REM PROGRAM 5.2
110 REM Enveloppes mat d'amplitude
120 :
130 ep=3200
140 FOR he=8 TO 15
150 PRINT he
160 ENV 1,=he,ep,5,0,100
170 SOUND 1,478,0,0,1
180 a$=INKEY$:IF a$="" THEN 180
190 NEXT he
```

La dernière enveloppe dans la définition ENV

produit un temps de pause de 5 secondes. Attendez 5 secondes avant d'appuyer une touche pour aller à la prochaine enveloppe. Changez **ep** en 130 et écoutez la différence. Comme **ep** détermine la vitesse des enveloppes, nous ne pouvons parler de leurs caractéristiques qu'en termes relatifs. Vous trouverez ci-dessous un résumé de leurs propriétés:

8 : attaque rapide, répétitive, déclin lent.
9 : attaque rapide, déclin lent, maintien au volume 0.
10: attaque rapide, déclin lent répétitif, attaque lente.
11: attaque rapide, déclin lent, attaque rapide, soutien au niveau maximal du volume.
12: attaque lente répétitive, déclin rapide.
13: attaque lente, soutien au maximum du volume.
14: attaque lente répétitive, déclin lent.
15: attaque lente déclin rapide, maintien au volume 0.

La valeur V, dans la commande SOUND est ignorée par une enveloppe matérielle.

Un effet de banjo peut être produit avec une valeur EP de 300 et une enveloppe de 8. Essayez également avec des valeurs plus faibles, inférieures à 20. Celles-ci tendent à épaissir le son. L'enveloppe 14 peut produire des effets de trémolo (voir chapitre 6 à ce sujet). Essayez les valeurs EP de 300 et plus.

Vous trouverez aisément des enveloppes matérielles s'adaptant à de nombreuses applications.

ENT : l'enveloppe de tonalité

Vous serez soulagé d'apprendre que la compréhension de l'enveloppe de tonalité est un

peu plus facile. Cette enveloppe est utilisée pour créer des effets étranges et merveilleux, également des sons musicaux très subtils.

De la même manière qu'ENV varie le volume d'une note pendant une période de temps, la commande ENT varie la fréquence de la note. La fréquence initiale sur laquelle agit l'enveloppe est déterminée par la valeur P de la commande SOUND. Le guide d'utilisation liste l'enveloppe logicielle de la manière suivante:

ENT,S,T1,V1,W1,T2,V2,W2,T3,V3,W3,T4,V4,W4,T5,V5,W5

S est le numéro de l'enveloppe ENT. A nouveau les paramètres T,V,W ne sont pas très explicites, ils portent les mêmes définitions que P,Q,R, mais leurs gammes diffèrent. Ils sont listés en figure 5.5. Si une commande ENT a été définie , l'AMSTRAD garde en mémoire toutes les caractéristiques jusqu'à l'arrivée d'une autre commande ENT. Il y a moyen de mettre à zéro une commande ENT en entrant la commande sans ses paramètres.

ENT 1

Paramètre	Nom	Spectre
S	Numéro d'enveloppe (– ve pour répétition)	1 à 15
Logiciel	S, T, V, W d'ENT	
T V W	Nombre d'intervalles Taille de l'intervalle Temps de pause	0 à 239 – 128 à 127 0 à 255
Matériel	S, = TP, PT d'ENT	
= TP PT	Période de tonalité Temps de pause	0 à 4 095 0 à 255

Figure 5.5. Les paramètres ENT et leur échelle de valeur.

T, V et W : UNE SECTION DE L'ENVELOPPE DE TONALITE

Chaque section T,V,W est auto-suffisante et l'on peut définir jusqu'à cinq sections semblables. Toutes ces sections travaillent de manière identique. Il faut obligatoirement définir une section complète, mais le nombre des sections est optionnel.

T désigne le nombre d'intervalles de la section. V représente la valeur de la variation de hauteur par intervalle. W spécifie la longueur de chaque intervalle en 1/100 de seconde.

Valeurs spéciales de S, T, V et W

Si S est négatif l'enveloppe de tonalité sera répétée. Le bit de répétition n'est pas effectif avec une valeur négative dans la commande SOUND comme il l'est avec une répétition de l'enveloppe de volume.

Si T=0, il agit comme si sa valeur était égale à 1.

Si V tente d'amener la hauteur en dehors des limites (0 à 4095), la hauteur opère un recouvrement comme l'a fait l'enveloppe de volume, c'est-à-dire qu'une hauteur qui dépasse le maximum reprendra au minimum.

Si V=0 alors cette section va maintenir la hauteur à sa valeur actuelle pour une durée égale à T x W.

La valeur W=0 est traitée comme si elle était égale à 256.

La durée d'une enveloppe de volume n'a pas d'effet sur la durée d'une note.

Rappelez-vous que plus la hauteur est faible, plus la fréquence est élevée. Ainsi une enveloppe de tonalité incrémentant la hauteur

produira des notes de plus en plus basses et vice-versa.

ETUDE DES ENVELOPPES DE TONALITE AVEC LE PROGRAMME GRAPHIQUE

Chargez et effectuez le programme 5.1. Si vous ne l'avez pas fait précédemment, appuyez sur TAB. Ceci vous passera de ENV à ENT. ENT1 est défini par une suite de zéros. Pressez I pour appeler l'enveloppe initiale.

Le fait d'appuyer sur la touche RETOUR vous affiche l'enveloppe sur l'écran. Comme il y a 4096 hauteurs possibles, l'axe des Y est divisé en 4096 sections. Evidemment, vu la résolution de l'écran, de faibles changements de hauteur ne pourront être distingués. Le fait de presser Y vous permet de changer l'échelle Y, mais attention à ne pas sortir de l'écran. Cette caractéristique a été incluse pour vous permettre de visualiser les petits changements de hauteur.

L'axe représenté sur l'écran est proportionnel à la hauteur de la note. Si vous pressez P et que vous entrez une hauteur proche de la limite inférieure ou supérieure vous pouvez observer l'effet de recouvrement.

R vous permet de passer du mode répétition au mode normal. Le mode répétitif est obtenu en programmant une enveloppe négative. L'écran nous affiche les variations de hauteur et non les fréquences produites. Par conséquent, une ligne montante représente une fréquence descendante.

Le temps d'ENT est affiché à l'écran. C'est le temps d'un cycle; il n'est pas valable en cas de répétition. Si ce temps est plus long que le temps d'ENV, son effet complet ne sera pas entendu. S'il est plus court et qu'il n'y a pas de répétition, la note va se terminer à une fréquence constante. Les graphiques ENT et ENV

ne seront pas dessinés à la même échelle. Je n'ai pas inclus des exemples de paramètres, la découverte de vos propres enveloppes et paramètres me semble plus profitable. La table des caractéristiques des instruments à la fin de ce chapitre vous donnera quelques idées d'expériences.

L'ENVELOPPE MATERIELLE ENT

Cette partie semble également plus facile à comprendre que l'enveloppe matérielle du volume. Elle prend la forme suivante:

=TP,PT

où TP est la période de tonalité ou hauteur et PT le temps ou durée de pause, identique à W dans l'enveloppe logicielle. De même les deux sections, matérielle et logicielle, peuvent être mélangées.

Ces enveloppes peuvent être définies pour jouer de petits accords tels que nous les trouvons dans le programme 1.5. Vous pouvez continuer aisément vos expérimentations.

Enveloppes multi-sections

Si vous avez suivi les commentaires sur le programme d'affichage des enveloppes, vous avez pu remarquer que le nombre de sections a été changé par une instruction POKE. Que se passe-t-il si nous spécifions, de cette manière, un nombre de sections supérieur à 5. Eh bien, l'AMSTRAD n'y voit aucun inconvénient, vous pouvez programmer des commandes ENV et ENT avec cent sections ou plus. Vous aurez du mal à trouver cela sur un synthétiseur du commerce.

Vous venez à peine d'en finir avec la mécanique d'une section et voilà que l'on vous lance dans des enveloppes à cent sections. Ne blêmissez pas trop. D'autant moins que quelques

pages plus tôt nous indiquions que quatre sections ADSR étaient suffisantes pour définir la tonalité d'un instrument de musique. Cependant, pour les plus avancés d'entre vous, ceux qui sont déjà familiers avec les cinq sections, vous avez ici l'occasion de créer des enveloppes extrêmement performantes.

Zone mémoire de stockage des enveloppes

Il y a trois positions-mémoire impliquées dans le stockage des définitions d'enveloppe et nous utiserons une notation hexadécimale pour nous adresser à elles. La notation hexadécimale est expliquée dans le guide d'utilisation, annexe 2. La description de chaque enveloppe occupe en mémoire 16 octets, ou en hexa &10.

La première définition d'une enveloppe est stockée dans la mémoire tampon, dans une zone allant de &ADBB à &ADCA. Les définitions de l'enveloppe de volume sont gardées dans les positions mémoires &B61A à &B709. La zone de stockage ENT est consécutive à la zone ENV.

Quand une enveloppe est définie, le contenu complet du segment tampon est transféré dans la zone mémoire, abstraction faite du nombre de sections contenu dans les enveloppes. La première adresse de chaque enveloppe contient le nombre de sections. Cette adresse peut être altérée avec un POKE, comme nous l'avons vu.

Les quinze octets restant sont divisés en cinq groupes de trois octets et contiennent les paramètres des sections. Vous comprendrez aisément que si vous écrivez en mémoire plus de cinq sections à l'aide de l'instruction POKE vous allez détruire le contenu de l'enveloppe suivante.

Les adresses mémoire peuvent être déterminées très facilement:

```
ENV1    &B61A - &B629
ENV2    &B62A - &B639
ENV3    &B63A - &B649
ENV4    &B64A - &B659 ...
ENV15   &B6FA - &B709
ENT1    &B70A - &B719
ENT2    &B71A - &B729
ENT3    &B72A - &B739 ...
ENT15   &B7EA - &B7F9
```

Le programme suivant affiche les adresses mémoire et nous montre comment les définitions sont stockées. Avant de lancer le programme, remettez à zéro la mémoire en arrêtant l'ordinateur.

```
100 REM PROGRAM 5.3
110 REM Zone de stokage de l'enveloppe
120 :
130 PRINT"BUFF  ";
140 FOR n=&ADBB TO &ADCA
150 IF n=&ADBB THEN PEN 3 ELSE PEN 1
160 PRINT PEEK(n);CHR$(8);
170 NEXT n
180 PRINT
190 :
200 FOR n=1 TO 3:REM Que 3 1ères ENVs
210 PRINT:PRINT"ENV";n;
220 FOR m=0 TO 15
230 IF m=0 THEN PEN 3 ELSE PEN 1
240 PRINT PEEK(&B61A+m+(n-1)*16);CHR$(8)
    ;
250 NEXT m
260 NEXT n
270 :
280 PRINT:PRINT
290 FOR n=1 TO 3:REM Que 3 1ères ENTs
300 PRINT:PRINT"ENT";n;
310 FOR m=0 TO 15
320 IF m=0 THEN PEN 3 ELSE PEN 1
330 PRINT PEEK(&B70A+m+(n-1)*16);CHR$(8)
    ;
340 NEXT m
350 NEXT n
360 SOUND 1,478,0,0,1,1
370 PRINT:PRINT
380 LIST 420-440
```

```
390 END
400 :
410 REM Enveloppes
420 ENV 1,1,15,10,2,-2,20,1,0,200,11,-1,
    10
430 ENV 1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0:
    ENV1
440 ENT 1,1,1,1
```

Commentaires

La ligne 160 affiche le contenu de la mémoire tampon. Les lignes 240 et 330 sont d'une approche un peu plus difficile, elles calculent les adresses-mémoire. En changeant les lignes 200 à 290, vous pouvez choisir l'enveloppe à afficher. Le nombre de sections dans chaque enveloppe est affiché en rouge. Le fait d'arrêter puis rallumer votre ordinateur, comme suggéré, chargera des valeurs aléatoires dans votre mémoire-tampon. Pour nettoyer votre mémoire tampon, entrez:

```
ENV1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0:ENV1
```

Ceci est une des lignes listées au bas de l'écran. Comme elles sont en dehors de la boucle du programme, elles ne seront pas exécutées. Utilisez le curseur et les touches d'édition pour les changer.

Copiez la première définition ENV1, puis exécutez le programme. Vous vous apercevrez que le segment et ENV1 contiennent les mêmes informations. Notez le stockage des paramètres négatifs. Tapez:

```
ENV2
```

Puis exécutez à nouveau: ENV2 contiendra les mêmes informations que la mémoire tampon et remarquez que le nombre de sections est maintenant zéro. Copiez ENT1. Pas de surprise. Maintenant entrez:

```
ENT -1,1,1,1
```

Le nombre de sections affiché sera de 129. En termes binaires cela veut dire que le bit de répétition 7 est activé, en plus des autres. Pour le prouver entrez juste la valeur de répétition:

```
ENT -1
```

et le nombre des sections passe à 128.

Stockage des enveloppes matérielles

Le premier paramètre de ce genre d'enveloppe est stocké comme un nombre à 16 bits et utilise donc deux octets consécutifs en mémoire. Nous examinerons l'enveloppe de tonalité. Entrez:

```
ENT 1,=478,20,=319,40
```

Le temps de pause est le même, mais la hauteur a été divisée en deux octets remplaçant les paramètres T (nombre d'intervalles) et V (taille de l'intervalle). Seules les valeurs supérieures à 239 sont acceptées pour le nombre d'intervalles T; vérifiez que T1 et T2 ont une valeur de 241.

La hauteur est stockée da la manière suivante: les valeurs au-dessus de 240 dans la colonne T, ajoutent 256 ou 2^8 à la hauteur. 478 est stocké de la manière suivante:

```
241,222 soit (241-240)*256+222
```

qui est équivalent à 478. C'est de cette manière que l'ordinateur stocke les nombres supérieurs à 255 sur plus d'un octet. Cela ne paraît pas très simple.

Pour découvrir la manière dont sont stockées les enveloppes matérielles de volume et pourquoi elles n'ont pas de temps de pause associé, vous pouvez commencer par comparer:

```
ENV1,=8,255
ENV1,=8,256
```

Augmenter le nombre de sections de l'enveloppe

Après avoir expérimenté le dernier programme vous aimeriez peut-être voir quel peut être le résultat lorsqu'on définit un mode avec enveloppe multi-sections. Le prochain programme implémente directement en zone mémoire ENV1, les paramètres P,Q,R.

Ajoutez la routine suivante à la fin du programme 5.3 et faites-le tourner en entrant:

```
RUN 1030
```

Vous verrez alors comment les valeurs ont été stockées en exécutant le programme original.

```
1000 REM PROGRAM 5.4
1010 REM Enveloppes multi-sections
1020 :
1030 nombre=0
1040 mem=&B61A
1050 WHILE k$<>"N"
1060 nombre=nombre+1
1070 PRINT "section";nombre
1080 INPUT "P (nombre d'intervalles)";ev
     (1)
1090 INPUT "Q (taille de l'intervalle)";
     ev(2)
1100 ev(2)=(256+ev(2)) MOD 256
1110 INPUT "R (temps de pause)";ev(3)
1120 FOR n=1 TO 3
1130 POKE(mem+n+(nombre-1)*3),ev(n)
1140 NEXT
1150 PRINT
1160 PRINT "Encore une autre section (O/
     N)?"
1170 k$=UPPER$(INKEY$):IF k$="" THEN
     1170
1180 PRINT
1190 WEND
1200 POKE(mem),nombre
```

Commentaires

La variable **mem** en 1040 est chargée avec l'adresse de début de la zone-mémoire, soit ENV1. Des adresses 1120 à 1140 nous trouvons les valeurs entrées au clavier à l'aide de l'instruction POKE.

La routine est très courte et aucune détection d'erreurs n'a été incluse, soyez donc très attentif ou incluez votre propre routine. La ligne 1100 met en forme la valeur Q puisqu'un octet ne peut contenir une valeur négative. Après avoir entré les sections, appuyez sur N, et la valeur sera écrite en mémoire.

Nous avons vu la manière d'utiliser des paramètres en dehors de la limite des enveloppes logicielles pour stocker des enveloppes matérielles. En utilisant ce programme vous pouvez entrer de telles valeurs, mais les résultats sont imprévisibles.

Une utilisation intéressante des enveloppes multi-sections est de déterminer une enveloppe matérielle de tonalité, avec une série de fréquences. Il vous faut convertir la hauteur en deux octets de la manière suivante:

```
dif=hauteurç256
high=240+dif
low=hauteur-dif*256
```

Ajoutez ces lignes au programme 5.4:

```
1015 REM PROGRAM 5.5
1016 REM Matériel multi-sections
1017 REM Enveloppes de tonalité
1018 REM Insérer ces lignes
1019 REM au PROGRAM 5.4
1020 :
1040 mem=&B70A
1080 INPUT "Hauteur";hauteur
1090 dif=hauteur\256
1100 ev(1)=240+dif
```

```
1105 ev(2)=hauteur-dif*256
1195 PRINT "Encore une fois (O/N)?"
1196 r$=UPPER$(INKEY$):IF r$="" THEN
     1196
1200 IF r$="O" THEN POKE(mem),hauteur+
     128 ELSE POKE(mem),nombre
```

Ne pas oublier de définir une enveloppe de volume adaptée pour entendre votre musique.

Expériences avec les programmes

Il y a un potentiel immense pour la musique dans les commandes ENV, en utilisant les définitions logicielles ou matérielles ou en programmant directement comme nous l'avons fait. Le programme 5.1 est très bien adapté pour vous familiariser rapidement avec les effets des paramètres. L'usage des enveloppes multi-sections permet d'économiser de la place-mémoire ainsi que du temps de programmation.

Caractéristiques des instruments

Pour vous aider dans vos futures expériences le programme 5.6 décrit les caractéristiques des instruments les plus courants, que vous pouvez convertir en paramètres d'enveloppes.

Gardez à l'esprit les limites du générateur de sons. Il est impossible de produire des imitations exactes, mais rappelez-vous que pour un même son, une personne vous dira que c'est une clarinette, l'autre un haut bois. En fait, l'interprétation dépend un peu de la personne. Les courbes d'enveloppe sont définies en termes d'attaque et de déclin. Dans ce contexte, le déclin se réfère à la phase de relâchement. La phase de déclin de l'ADSR suivra des lignes similaires à celles de l'échantillon d'enveloppe d'instrument de la figure 5.2. Les cinq phases

d'enveloppe de l'AMSTRAD ajoutent une corde à votre arc, c'est-à-dire que cette enveloppe à multiples sections vous libère des limites quelquefois imposées par les quatre sections de l'ADSR.

N'oubliez pas non plus que vous pouvez créer une variété de sons complètement nouveaux. Ceci est un des aspects les plus intéressants et créatifs de la musique sur AMSTRAD.

Avec l'addition de quelques effets, et la souplesse de l'enveloppe de tonalité, vous pouvez créer un micro-orchestre des plus acceptables. Le chapitre suivant traite de la création et de l'utilisation de tels effets. Nous nous référons à ces effets en tant qu'ornements musicaux.

Instrument	Octaves	Attaque	Déclin	Applications et effets spéciaux
Violon Viole Violoncelle	0 à 2 – 1 à 1 – 2 à 0	lente lente lente	moyen/lent moyen/lent moyen/lent	portamento vibrato gammes
Basse	– 2 à 0	rapide	lent	figures de rythme
Trompette	0 à 2	rapide	moyen/rapide	vibrato
Trombone	– 2 à 0	moyenne/rapide	moyen/rapide	portamento
Tuba	– 3 à – 1	lente	moyen/rapide	pour souligner les staccato
Saxophone alto Saxophone ténor	– 1 à 1 – 2 à 0	moyenne/rapide moyenne/rapide	moyen/rapide moyen/rapide	attire la note altère la note
Flûte	0 à 2	moyenne/rapide	moyen/rapide	trilles, arpèges
Clarinette Hautbois	– 1 à 1 0 à 2	moyenne/rapide moyenne/lente	moyen/rapide moyen/rapide	vibrato lent
Basson	– 2 à 0	rapide	moyen/rapide	lent, lugubre
Guitare	– 2 à 1	rapide	lent	tire la note
Accordéon	– 2 à 2	moyenne/lente	moyen/rapide	légèrement faux
Harpe	– 3 à 3	rapide	lent	arpèges, glissando
Xylophone	0 à 3	rapide	moyen/rapide	glissando
Caisse claire	moyen	rapide	moyen	roulements de tambour
Grosse caisse	bas	rapide	moyen	battements courts
Orgue	– 3 à 3	rapide	rapide	vibrato, accords complets

Figure 5.6.

6

Variations musicales

Un synthétiseur possède plusieurs options pour améliorer et embellir les sons générés. Quelques effets en résultant sont complexes, plusieurs sont subtils, d'autres non. Le musicien peut acheter une multitude de pédales et de boîtes pour produire des échos, des réverbérations, du fading, du "phasing"...la liste n'en finirait plus ! Dans ce chapitre, nous allons voir quels effets l'AMSTRAD peut produire.

Vibrato et trémolo : modulations de fréquence et d'amplitude

Généralement ce sont les premiers effets qu'un musicien apprend à incorporer dans un morceau. Ils font partie des sons naturels, et ajoutent vraiment une nouvelle dimension d'entrain aux ondes sinusoïdales ou rectangulaires. Il existe de très nombreuses confusions, même chez les musiciens; nous allons donc tout de suite les dissiper.

Le vibrato est une modulation de fréquence, le trémolo une modulation d'amplitude. La modulation est généralement régulière et consiste en une augmentation et une diminution, de la fréquence de la note, ou de son volume. Une représentation graphique de ces variations serait similaire à la courbe visualisée au chapitre 1.

Il n'y a que peu d'instruments qui produisent le trémolo. Les orgues électroniques produisent cet effet mécaniquement et électroniquement; les chanteurs aussi utilisent cet effet pour embellir leur voix. Le vibrato est plus commun et souvent utilisé par les musiciens sur les instruments à cordes, les cuivres et les clarinettes.

La manette de trémolo des guitares électriques, intensivement utilisée dans les années 60, génère en réalité un vibrato. En tirant et en relâchant les cordes, la fréquence augmente puis diminue. Si cette manette est manipulée rapidement, l'effet produit est un vibrato régulier; dans le cas contraire, on arrive à produire un portamento. La modulation la plus agréable du vibrato est obtenue autour d'une variation de sept cycles par seconde pour le trémolo, les variations allant du subtil au ridicule.

Vous avez probablement découvert de nombreux exemples dans des jeux et vous-même en avez inventé quelques-uns dans le programme du dernier chapitre. Le vibrato étant le plus commun des deux, nous allons l'étudier en premier.

Création de vibrato avec la commande ENT

La modulation de fréquence de vibration du son dans un vibrato, fréquence que nous avons appelée "hauteur" jusqu'à ce point de notre ouvrage, est souvent inférieure à un demi-ton. Dans un vibrato musical, la fréquence varie régulièrement autour de la fréquence de base de la note, en dessinant une onde sinusoïdale. Des exemples extrêmes, où la fréquence varie très rapidement et dans des limites beaucoup plus grandes, peuvent être produits par des moyens électroniques pour des effets spéciaux et sont appelés vibratos "techniques".

```
10 ENT -1,20,10,1,40,-10,1,20,10,1
20 SOUND 1,478,500,7,0,1
```

En changeant la vitesse et la modulation nous pouvons altérer très fortement le son et produire d'innombrables effets qui n'ont qu'un rapport lointain avec la musique. Pour produire un vibrato avec la commande ENT, nous pouvons utiliser la formule empirique suivante:

```
ENT -1,X,D,S,X,-D*2,S,X,D,S
```

Le degré de vibrato dépend également de la fréquence de base de la note ainsi que de l'ampleur de la modulation de fréquence. Vous avez ci-dessous quelques exemples d'enveloppe de vibrato. Vous pouvez utiliser le programme 5.1 pour comparer. Mettez **yscale** à 1:100

```
ENT -1,3,1,2,6,-1,2,3,1,2
ENT -1,3,1,2,3,-2,2,3,1,2
ENT -1,1,1,1,1,-2,1,1,1,1
ENT -1,1,1,6,1,-2,6,1,1,6
```

Vous aurez vite repéré quel effet vibrato vous convient le mieux musicalement.

Production d'effets trémolo

Pour produire des trémolos, il suffit de varier le volume dans une fréquence sinusoïdale, de la même manière que pour le vibrato. Nous pouvons faire cela en jouant sur le paramètre D de la commande SOUND, en lui affectant une valeur négative. La valeur absolue de ce nombre stipulera la valeur de répétition de l'enveloppe de volume. Malheureusement, nous perdons tout le contrôle de la durée de la note.

Pour l'exemple suivant, nous allons mettre le volume initial à 7, dans la commande SOUND. Si vous utilisez le programme 5.1, modifiez ces lignes :

```
3560 IF com=13 THEN SOUND 129,hauteur, -10,7,1,1
3620 SOUND 1,note,-dur,7,1,1
3110 MOVE 0,112
```

La ligne 3130 déplace le graphique pour pouvoir visualiser les variations de volume avec un son initial à 7. Après avoir fait RUN, mettez l'enveloppe suivante:

```
ENV 1,8,1,3,14,-1,3,6,1,3
```

Vous obtiendrez un bon trémolo. Chaque note est répétée vingt-cinq fois et a une durée de 25 x 0,56 secondes. Changez l'enveloppe en:

```
ENV 1,8,1,1,14,-1,1,6,1,1
```

et mettez la durée à 1 puis à 2. Cela semble mieux, mais de cette manière vous pouvez voir les problèmes inhérents à la production de trémolo. Je n'ai pu produire une très bonne enveloppe de trémolo, mais la série de niveaux produite semble suffire pour certaines applications. Dans le trémolo ci-dessus, le volume passe du maximum au minimum. Pour des effets plus subtils, réduisez l'étendue des variations de volume, pour ne plus passer que par quatre ou six niveaux.

L'enveloppe matérielle n 14 avec un EP de 500 ou plus produit un trémolo sans effet externe.

Si vous ajoutez des enveloppes de tonalité à ces expériences, vous découvrirez des sons intéressants simulant le fading en plus du trémolo.

Mais attention à la saturation, trop ajouter d'embellissements nuit. Si un morceau de musique ne contenait que des vibratos et des trémolos, ce ne serait plus agréable du tout.

Le vibrato, vous le verrez, est très utile musicalement, et très facile à utiliser; quant au trémolo, on a encore mal cerné tout son potentiel, et les effets qu'il produit peuvent être assez bizarres.

Les trilles : un vibrato spécial

Pour produire un vibrato sur un instrument, il faut pouvoir contrôler la hauteur entre deux notes. Les notes du piano, par exemple,sont fixes et il vous est impossible de produire un vibrato sur cet instrument. La seule chose que vous puissiez faire est d'alterner rapidement le jeu de deux notes adjacentes; ceci est appelé un trille. Exécutez la version non modifiée du programme 5.1 et définissez l'enveloppe suivante:

```
ENT -1,1,-105,8,1,105,8
```

Si vous appuyez sur la touche RETOUR, vous produirez un trille entre le DO et le RE, respectivement de l'octave inférieure et supérieure au DO central du piano. En sautant d'une note à l'autre, la hauteur ne suit pas une courbe sinusoïdale comme elle le faisait pour le vibrato et le trémolo, mais plutôt une forme rectangulaire. Un trille est une oscillation entre notes distinctes et peut donc être exécuté sur la plupart des instruments. Si nous voulons inclure un trille dans un morceau de musique nous pouvons indiquer à l'ordinateur de jouer ces deux notes, mais souvent il est plus intéressant de passer le contrôle à une enveloppe qui prendra la place des seize instructions ou plus qui définissent la hauteur.

Un trille peut être joué en utilisant n'importe quel intervalle. Il faut garder à l'esprit, cependant, la différence de hauteur entre les deux notes : la hauteur produite par une enveloppe de trille ne produira une vraie note que si elle est utilisée pour la hauteur pour laquelle elle a été conçue. Heureusement, notre oreille n'est pas trop sensible et s'accommode assez bien de trilles joués sur un ou plusieurs octaves. Vous trouverez des exemples dans le programme suivant:

```
100 REM PROGRAM 6.1
110 REM Introduction de "Musique
    Militaire"
120 :
130 ENT -1,1,-7,8,1,7,8
140 SOUND 1,60,270,6,0,1
150 FOR n=1 TO 9
160 READ hauteur,dur
170 SOUND 4,hauteur,dur,7
180 NEXT n
190 END
200 DATA 478,60,506,20,536,60,568,20,638
    ,20,716,20,758,20,851,20,956,30
```

ECHO ECHo ECho Echo echo ET REVERBERATION

Ces deux effets sont souvent utilisés sans poids ni mesure, plus que tout autre de l'arsenal d'un synthétiseur. C'est qu'en fait ils sont très amusants, mais si on sait bien les manipuler, ils peuvent produire quelque chose de très beau.

Les échos sont produits quand les ondes se réfléchissent sur des surfaces lisses et dures, telles que des falaises. Si vous vous mettez à crier devant une falaise, ou un escarpement rocheux, l'onde sonore produite par votre voix frappe le mur et rebondit. Le décalage dépend de votre distance par rapport au mur. Pour pouvoir distinguer l'onde réfléchie, il faut au moins un temps de retard de 1/10 de seconde, soit une distance d'environ 16 m. Plus près, l'écho n'est pas mis en valeur.

Un son émis dans une chambre rebondit sur les quatre murs, le sol et le plafond, l'écho résultant est complexe et fait de multiples réflections. Le son est renforcé et semble continuer après l'arrêt du son initial. Cette forme d'écho est appelée réverbération. Le temps de réverbération est le temps mis par un son réfléchi pour tomber à un certain niveau. Les pièces ayant des surfaces vitrées ou carrelées ont un temps de réverbération plus important que

les pièces entièrement moquettées. Cela explique peut-être pourquoi autant de personnes préfèrent chanter dans la salle de bain ...

Les unités de réverbération du commerce produisent leurs effets au moyen de ressorts et de plaques métalliques. Il faut générer un très grand nombre de vibrations et l'AMSTRAD n'en a pas les moyens. Rien ne vous interdit de brancher la sortie de votre ordinateur sur une unité de réverbération. La meilleure chose que nous puissions faire est de reproduire une phase de relâchement très très longue dans l'enveloppe. Par contre, nous pouvons produire des échos.

Production d'échos avec les unités du commerce

L'écho est simplement la répétition d'une note. Les unités d'écho sont capables de produire de un à cinq, voire six échos. De nombreuses unités peuvent en provoquer beaucoup plus; quelques-unes, particulièrement les unités digitales, sont capables de créer un nombre infini d'échos. Il faut être conscient que dans un écho naturel le son réfléchi est plus faible que le son émis. Si un écho est répété plusieurs fois, il y a un seuil à partir duquel le son devient inaudible. La plupart des unités vous permettent de contrôler le volume sonore de l'écho réfléchi, pour pouvoir choisir entre une diminution rapide ou lente de l'écho.

Un autre type d'unité, que l'on retrouve surtout dans le milieu des chanteurs de variété, est le magnétophone équipé de têtes multiples. La première tête récupère l'enregistrement, la seconde superpose le même son une fraction de seconde plus tard, la troisième récupère etc.

Deux techniques différentes couvrent le marché, la technique analogique et la technique digitale. Les unités analogiques contiennent des circuits qui retardent le signal pendant un certain temps, pour le passer au circuit suivant qui en fait autant. Cette forme d'écho a

évidemment de nombreux inconvénients. Le son passant d'un circuit à un autre, la qualité décroît rapidemment et il y a des limites dans la reproduction des échos. Les unités digitales travaillent de manière similaire, mais les sons se trouvent transformés en une série de nombres binaires et ne peuvent donc être altérés.

La production d'échos sur AMSTRAD

Pour produire un bon écho, il faut un intervalle perceptible entre les deux moments où le son diminue et augmente à nouveau. Nous pouvons programmer cet effet jusqu'à un certain point. En utilisant le programme 5.1, définissez cette enveloppe-ci :

```
ENV 1,18,5,10
```

Vous allez vous apercevoir que nous avons la production d'un écho correct, sauf pour la dernière phase. Si nous incrémentons le premier paramètre à 19 le cycle recommence, mais le niveau sonore est à nouveau mis à 15. Ceci est dû à la nature de l'enveloppe que nous avons définie. Nous pouvons ajouter un relâchement lent à la fin de l'enveloppe pour émettre une pseudo-réverbération:

```
ENV 1,18,5,10,10,-1,15
```

Ceci peut suffire à la plupart des applications. Vous pouvez changer la vitesse de l'écho, en jouant sur le paramètre R. Nous pouvons utiliser tous les paramètres d' ENV.

```
ENV 1,1,15,17,8,7,17,1,-7,17,1,5,17,2,-2,17
```

L'écho semble correct. A nouveau le paramètre R contrôle la vitesse de l'écho. Incrémentez le dernier paramètre pour allonger l'effet d'écho. Pour vraiment définir une enveloppe d'écho, vous pouvez utiliser les enveloppes multi-sections, telles que nous les avons décrites dans le dernier chapitre. Vous aurez le contrôle du laps de temps individuel,

ainsi qu'un contrôle absolu du volume de chaque écho.

Divers effets spatiaux

Nous avons regroupé ici plusieurs effets spéciaux, bien que chacun ait son caractère propre. Les sonorités sont produites en délayant un son et/ou en variant sa fréquence. Ils sont appelés effets spatiaux car ils déforment le son par des effets de réverbération et nous les percevons comme des sons réfléchis dans des salles immenses.

L'effet chorus

Quand un groupe de musiciens joue à l'unisson, chacun joue en vérité à une fréquence légèrement différente. Cette différence est minime, mais c'est elle qui permet à notre oreille de distinguer les instruments entre eux. Ceci est appelé l'effet chorus; il est mis à profit par les petits orchestres pour donner de l'ampleur à leur prestation. Les unités de chorus produisent ces effets en déformant légèrement la fréquence et en la recombinant avec la fréquence du son original.

Le phasing et le flanging

Ce sont des effets difficilement traduisibles en mots. Ils produisent généralement des sons utilisés dans la musique électronique. En décollant, un avion à réaction crée un effet de phasing.

Dans le milieu musical, il existe plusieurs versions sur la découverte du flanging. Une première version explique que le premier flanging a été produit en appliquant une légère pression sur les flancs d'une bande magnétique pendant sa lecture, pour créer un retard. Si vous pouvez imaginer le même effet avec deux bobines se déroulant en parallèle, vous vous représentez assez bien cet l'effet (vous avez

surtout une excellente imagination...)

Tous ces effets sont produits par diverses formes de retards qui sont toujours de l'ordre du 1/10 ou 1/100 de seconde. Quand un son est retardé puis mélangé au son original certaines fréquences sont supprimées. Si le retard change, la gamme des fréquences supprimées varie également et on arrive à produire un effet de glissement.

Nous arrêtons ici la description de ces effets, il est pratiquement impossible de retraduire ces effets sur le papier et je vous propose de les écouter pour mieux les apprécier. Peut-être que le vendeur de votre magasin de musique voudra bien vous faire une démonstration.

Les effets de retard sur AMSTRAD

Si nous programmons la même hauteur dans deux ou plusieurs canaux, la fréquence du son sera exactement la même. Si la fréquence d'un canal est changée très faiblement nous pouvons arriver à produire un effet de chorus. Nous pouvons même utiliser les trois canaux en différenciant légèrement les hauteurs. Voici un exemple simple de cette utilisation dans une imitation d'accordéon:

```
100 REM PROGRAM 6.2
110 REM Musique d'accordéon
120 REM Effet chorus
130 :
140 FOR air=1 TO 14
150 READ note,dur
160 SOUND 1,note,dur,7
170 SOUND 2,note-1,dur,6
180 NEXT air
190 END
200 DATA 426,48,338,48,201,48,213,432
210 DATA 426,48,319,48,201,48,213,240,
    268,48
220 DATA 239,72,402,24,426,24,478,24,
    426,24
```

Vous pouvez changer la valeur soustraite en 170. Vous préférerez peut-être l'effet produit par un soustraction de 3 ou 4.

Les battements de fréquences

Quand vous écoutez deux notes très proches l'une de l'autre quant à la hauteur,vous entendez leurs fréquences s'entrelacer. On appelle cela un battement. La fréquence de battement est égale à la différence des deux fréquences. Par exemple si deux fréquences sont produites, l'une à 440 Hz, l'autre à 441 Hz, la fréquence de battement sera de 1 Hz. Il est difficile de produire des fréquences exactes avec l'AMSTRAD, vu la précision de la formule de conversion fréquence/hauteur. Pour illustrer:

```
10 SOUND 1,284,1000,7
20 SOUND 1,283,1000,7
```

Nous pouvons également en faire l'expérience avec des enveloppes matérielles. Comme la période de l'enveloppe est un multiple de 128 microsecondes, des effets étranges sont obtenus en utilisant de faibles valeurs EP. Dans le programme 6.2, ôtez la REM en ligne 170 et changez la ligne 160 en:

```
160 SOUND 1,note,dur,0,1
```

Définissez une enveloppe matérielle répétitive avec:

```
ENV 1,=14,1
```

Puis changez le paramètre EP. L'effet varie suivant la hauteur de la note; dans les valeurs faibles, nous obtenons un vague son de guitare; plus loin, quand le paramètre atteint les valeurs 50, le son produit ressemble plus à une enveloppe de volume.

Les autres effets spatiaux sont vraiment au-dessus des possibilités de l'AMSTRAD mais nous pouvons satisfaire notre esprit en disant

qu'ils sont juste un peu plus sophistiqués que l'effet de chorus.

Le modulateur de résonance : tintements et sonneries

Le dernier module du synthétiseur que nous allons examiner et essayer de reproduire est le modulateur de résonance. Il permet d'émettre des sons métalliques ou de sonnettes. Il travaille différemment des autres modules.

Un modulateur de résonance demande une entrée de deux fréquences et génère un signal de sortie composé de la somme des deux fréquences et de leur différence. Pour prendre un exemple: avec les fréquences 440 Hz et 1220 Hz, le signal de sortie est composé d'un signal de 1660 Hz (440Hz+1220Hz) et d'un signal de 780Hz (1220Hz-440Hz).

Un tel signal a une forme d'onde dont les harmoniques ne sont pas apparentés les uns aux autres. Il est utilisé pour simuler des sonnettes, des gongs et autres instruments métalliques.

Les clochettes sur l'AMSTRAD

L'AMSTRAD ne possède pas un modulateur de clochettes, mais il peut reproduire des sons similaires en manipulant les fréquences. L'exemple suivant vous permet de suivre une telle expérimentation.

```
100 REM PROGRAM 6.3
110 REM Essais de sonnettes
120 :
130 ENV 1,1,15,6,15,-1,8
140 :
150 INPUT "Première fréquence";freq1
160 INPUT "Deuxième fréquence";freq2
170 CLS
180 LOCATE 6,8:PRINT"Freq 1:";freq1
190 LOCATE 25,8:PRINT"Freq 2:";freq2
```

```
200 LOCATE 1,10:PRINT"Freq sonnette 1:"
210 LOCATE 20,10:PRINT"Freq sonnette 2:"
220 GOSUB 370
230 :
240 WHILE -1
250 com$=INKEY$:IF com$="" THEN 250
260 com=ASC(com$)
270 IF com=59 THEN freq2=freq2+1
280 IF com=47 THEN freq2=freq2-1
290 IF com=97 THEN freq1=freq1+1
300 IF com=122 THEN freq1=freq1-1
310 LOCATE 13,8:PRINT freq1
320 LOCATE 32,8:PRINT freq2
330 GOSUB 370
340 WEND
350 :
360 REM Calcul des fréquences
370 sonnette1=freq1+freq2
380 sonnette2=ABS(freq1-freq2)
390 hauteur1=ROUND(125000/sonnette1)
400 hauteur2=ROUND(125000/sonnette2)
410 LOCATE 13,10:PRINT sonnette1
420 LOCATE 32,10:PRINT sonnette2
430 :
440 SOUND 129,hauteur1,1000,0,1
450 SOUND 130,hauteur2,1000,0,1
460 RETURN
```

Commentaires

La ligne 130 définit une enveloppe adéquate pour produire un son de sonnette, les lignes 150 et 160 demandent deux fréquences. Essayez 440 et 1220 Hz pour démarrer. La somme et la différence sont calculées des lignes 370 à 400. Les lignes 440 à 450 mettent en oeuvre les canaux A et B. Vous pouvez augmenter ou diminuer les fréquences en utilisant les touches A et Z pour la fréquence 1 et ";" et "/" pour la fréquence 2. Il se peut que vous ayiez le message : "Improper argument" (qui signifie "Argument ne convenant pas). Cela provient du fait que les combinaisons sont en dehors des limites du générateur de son. Vous pouvez inclure des lignes pour imprimer les fréquences 1 et 2.

L'AMSTRAD n'est pas équipé du même modulateur que les grands synthétiseurs, mais les résultats sont similaires. L'exemple suivant nous montre de quelle manière un modulateur de résonance peut être utilisé dans un accord.

```
100 REM PROGRAM 6.4
110 REM Sonnette modulant un air
120 :
130 ENV 1,1,14,6,14,-1,8
140 :
150 FOR stone=1 TO 25
160 PRINT:PRINT stone;"Elévation d'un
    demi-ton"
170 RESTORE
180 FOR tune=1 TO 13
190 READ note,dur
200 :
210 freq1=125000/note
220 freq2=2^(stone/12)*freq1
230 REM freq2=freq1/2^(stone/12):REM
    Descend
240 sonnette1=freq1+freq2
250 sonnette2=ABS(freq1-freq2)
260 note1=ROUND(125000/sonnette1)
270 note2=ROUND(125000/sonnette2)
280 :
290 PRINT"note1=";note1;" note2=";note2
300 SOUND 1,note1,dur,1,1
310 SOUND 2,note2,dur,0,1
320 NEXT tune
330 NEXT stone
340 END
350 DATA 190,48,213,48,159,48
360 DATA 142,24,106,24,127,48,142,24,106
    ,24,127,48
370 DATA 159,48,142,48,190,48,213,48
```

Commentaires

Ce programme joue l'accord vingt-cinq fois. Chaque fois, la fréquence est incrémentée d'un demi-ton. Si vous êtes capable de supporter les vingt-cinq interprétations sans une grimace vous n'êtes pas très délicat. Le but du programme

est de vous faire entendre des combinaisons de fréquences pour choisir le meilleur effet. Le programme prend les notes sous forme de hauteur pour les convertir en fréquences. La formule n'est pas toujours exacte, mais elle est suffisante pour nos besoins. La ligne 220 calcule la seconde fréquence. Si dans la ligne 230 vous supprimez la REM et lui substituez par la ligne 220, les demi-tons vont baisser. Les calculs sont similaires à ceux du programme 6.3.

L'enveloppe maintient le niveau à 14. La commande SOUND, en ligne 230 sera un peu plus faible que celle de la ligne 300.

Normalement, la fréquence originale n'apparaît pas à la sortie du modulateur. Ajoutez la hauteur originale et écoutez le résultat :

```
315 SOUND 4,note,dur,0,1
```

Les effets de sonnerie ne sont pas souvent utilisés dans la musique synthétique ou sur l'AMSTRAD, mais vous trouverez peut-être que le sujet vaut la peine d'être exploré.

7

Bruitages, sifflements, et autres bizarreries

Le septième paramètre de la commande SOUND, N, que nous avons à peine mentionné, contrôle le générateur de bruit (anglais "noise"). Sans lui, vos jeux sur l'AMSTRAD n'auraient ni "bang" ni "crash". Nous allons voir comment créer des effets un peu plus subtils dans d'autres types de programmes. Le bruit, c'est... du bruit! Mais il y a plusieurs sortes de bruits. Le bruit produit par le synthétiseur de l'AMSTRAD est un bruit pseudo-aléatoire qui ressemble énormément au bruit blanc.

Le bruit blanc

Tous les circuits électriques génèrent des bruits, souvent indésirables. Dans les synthétiseurs cet effet peut être utilisé de différentes manières: comme source de son sans hauteur bien précise ou comme partie non déterminée d'un son ayant une hauteur connue.

Le bruit blanc est un mélange à parts égales des fréquences auditives, de la même manière que la "couleur" blanche est une recomposition de toutes les couleurs.

Si nous changeons d'octave en passant du DO0 au DO1 par exemple, la fréquence de la note double. Il y aura donc des fréquences plus élevées dans les octaves hautes que dans les octaves basses. Le sifflement caractéristique du bruit blanc est dû aux fréquences élevées qui le composent.

Il y a d'autres formes de bruit. Le plus commun étant le bruit rose qui contient une quantité égale de fréquences de toutes les octaves. Il est similaire au bruit blanc, il ne contient tout simplement pas les fréquences les plus élevées. Il est très intéressant pour produire des bruits de vagues. Vous pouvez également produire du bruit rouge en augmentant le nombre de fréquences élevées que l'on élimine. D'ailleurs, sur tous les grands synthétiseurs il est possible de choisir la composition du bruit que l'on veut produire. Le bruit produit par l'AMSTRAD est fonction du paramètre N que nous avons vu dans le Chapitre 3.

Voici deux manières d'utiliser le bruit:

1- Il peut être utilisé tout seul en contrôlant la hauteur du son pour produire des bangs ou des explosions. Avec quelques variations nous pouvons même produire des effets de rythme.

2- Il peut être conjugué avec d'autres sons.

Effets sonores simples

Le programme 7.1 contient cinq exemples d'effets sonores. Vous avez la possibilité de rentrer au clavier le programme en entier, ou section après section. Chaque section possède sa propre enveloppe.

```
100 REM PROGRAM 7.1
110 REM Effets bruitage
120 :
130 REM Mitrailleuse
140 FOR explosion=1 TO 3
150 FOR balle=1 TO 12
160 SOUND 1,0,5,7,0,0,13
170 SOUND 1,0,5,7,0,0,18
180 NEXT balle
190 FOR d=1 TO 1500:NEXT
200 NEXT explosion
210 :
```

```
220 REM Ricochet
230 ENV 1,1,15,5,5,-1,10,10,-1,5
240 ENT 1,1,-120,1,40,1,3
250 FOR coup=1 TO 3
260 SOUND 1,150,100,0,1,1,15
270 NEXT coup
280 FOR d=1 TO 3500:NEXT
290 :
300 REM Cymbales ou Enclume
310 ENV 2,1,15,5,5,-1,10,10,-1,5
320 FOR clang=1 TO 8
330 SOUND 1,16,0,0,2,0,15
340 NEXT clang
350 FOR d=1 TO 6000:NEXT
360 :
370 REM Quelqu'un qui dort
380 ENV 3,7,2,10,5,-1,12,9,-1,6
390 ENT 4,6,2,1,12,-1,20,1,0,100
400 FOR ronflement=1 TO 5
410 SOUND 1,0,200,0,3,0,28
420 SOUND 1,0,0,0,4,0,31
430 NEXT ronflement
440 :
450 REM Soucoupe volante qui décolle
460 ENV 5,1,15,2,15,-1,50
470 ENT -2,1,2,2,1,-4,2,1,2,2
480 sifflement=46
490 FOR moteurs=31 TO 2 STEP -1
500 sifflement=sifflement-1.25
510 SOUND 1,sifflement,(moteurs/2)^2,15,
    0,2,moteurs
520 NEXT moteurs
530 SOUND 1,sifflement,0,0,5,2,2
```

Commentaires

Le bruit de mitrailleuse alterne entre deux hauteurs de bruit blanc. Ce principe peut être utilisé dans d'autres cas, pour produire de multiples effets. Otez REM en ligne 190 et changez le paramètre N en 160 et en 11, à la ligne 170. Ajoutez un autre retard pour produire le son des pales d'un hélicoptère.

```
175 FOR d=1 TO 90:NEXT
```

Otez REM en 190 et ajoutez la ligne suivante pour simuler une voiture qui a quelques problèmes de démarreur:

```
174 SOUND 1,0,5,7,0,0,18
```

Enlevez le paramètre D dans la commande SOUND pour rendre cette voiture encore plus hésitante.

Le ricochet est produit en combinant le bruit avec une baisse de hauteur produite par une enveloppe de tonalité. Le son produit par la hauteur toute seule ressemble plus à une mouette, encore que... Vous aurez plus d'informations à ce sujet un peu plus loin.

La routine "cymbales" est similaire à la routine "ricochet" par l'association bruit/son. Généralement les synthétiseurs produisent une sonorité de cymbale avec un petit claquement de bruit blanc, mais aussi avec un tintement métallique comme ici. Le résultat n'est pas parfait et ressemble beaucoup plus au bruit émis par une cymbale électronique. Si vous réduisez la durée de cet effet vous obtiendrez des cliquetis métalliques. Redéfinissez ENV2 en ligne 310.

```
310 ENV2,1,15,1,5,-1,2,10,-1,2
```

La routine du monsieur qui dort est un essai de ronflement. Avec un tout petit peu d'imagination, vous entendrez ronfler un géant.

La routine de soucoupe volante combine à nouveau son et bruitage. Quelques essais ont été nécessaires pour arriver à produire le gémissement des moteurs. La formule retenue est celle de la ligne 510.

Etude d'une unité rythmique

Après avoir expérimenté les différentes formes de bruits, armes à feu, véhicules..., il

est aisé de produire des bruits pour une unité rythmique. Le problème principal, et celui demandant le plus d'attention est la génération du bruit de la batterie Un excellent bruit de batterie demande une application toute particulière dans le choix des hauteurs et de l'enveloppe de volume. Essayez le modulateur de résonance pour vos bruits de percussions métalliques et comparez-le au bruit de cymbale obtenu au dernier exemple.

Vous pouvez vous apercevoir que la durée est un paramètre déterminant dans le choix des caractéristiques d'un instrument de percussion. Il vous faudra choisir plus d'une enveloppe pour produire une unité rythmique un peu sérieuse. Le prochain programme est une approche de l'étude des programmes rythmiques.

```
1000 REM PROGRAM 7.2
1010 REM Rhythme
1020 :
1030 GOSUB 1260:REM Titre
1040 GOSUB 1400:REM Mise au point
1050 :
1060 WHILE -1
1070 READ batterie,dur
1080 WHILE batterie<>0
1090 com$=INKEY$:IF com$>"" THEN temps$=
     com$
1100 SOUND 1,hauteur(batterie),dur*tempo
     ,0,ev(batterie),et(batterie),bruit(
     batterie)
1110 READ batterie,dur
1120 WEND
1130 IF temps$="1" THEN RESTORE 1820
1140 IF temps$="2" THEN RESTORE 1840
1150 IF temps$="3" THEN RESTORE 1870
1160 IF temps$="4" THEN RESTORE 1890
1170 IF temps$="5" THEN RESTORE 1920
1180 IF temps$="6" THEN RESTORE 1940
1190 IF temps$="7" THEN RESTORE 1960
1200 IF temps$="8" THEN RESTORE 1980
1210 IF temps$="9" THEN RESTORE 2010
1220 WEND
1230 END
```

```
1240 :
1250 REM Titre
1260 MODE 1
1270 LOCATE 9,3:PRINT "R H Y T H M E"
     :PRINT
1280 PRINT " 1) Rock 1"
1290 PRINT " 2) Rock 2"
1300 PRINT " 3) Rock 3"
1310 PRINT " 4) Mexican Rock"
1320 PRINT " 5) Insertion 1"
1330 PRINT " 6) Insertion 2"
1340 PRINT " 7) Insertion 3"
1350 PRINT " 8) Swing"
1360 PRINT " 9) Cha Cha"
1370 RETURN
1380 :
1390 REM Mise au point
1400 ENV 1,1,15,2,5,-3,4
1410 ENV 2,1,15,1,15,-1,2
1420 ENV 3,1,15,1,4,-2,2,7,-1,4
1430 ENV 4,7,2,1,14,-1,4
1440 ENV 5,1,15,1,15,-1,8
1450 ENT 1,1,-100,1,5,25,3
1460 ENT -2,1,-75,1,4,25,3
1470 ENT -3,1,4,2,1,-8,2,1,4,2
1480 :
1490 DIM hauteur(11),ev(11),et(11),bruit
     (11)
1500 REM percussions
1510 FOR n=1 TO 11
1520 READ hauteur(n),ev(n),et(n),bruit
     (n)
1530 NEXT n
1540 :
1550 REM 1 Tom1
1560 DATA 270,1,2,2
1570 REM 2 Tom2
1580 DATA 340,1,2,8
1590 REM 2 Tom3
1600 DATA 390,1,2,15
1610 REM 4 Tom4
1620 DATA 440,1,2,20
1630 REM 5 Grosse Caisse
1640 DATA 900,1,1,31
1650 REM 6 Caisse claire
1660 DATA 180,3,1,13
```

```
1670 REM 7 Cymbales Charleston
1680 DATA 0,4,0,1
1690 REM 8 Cymbales
1700 DATA 16,2,0,1
1710 REM 9 Baguettes
1720 DATA 0,1,0,15
1730 REM 10 Guiro
1740 DATA 20,5,3,15
1750 REM 11 Clochette
1760 DATA 80,1,0,10
1770 :
1780 tempo=4:temps$="1"
1790 RETURN
1800 :
1810 REM Rock 1
1820 DATA 1,3,1,3,6,3,6,3,5,6,5,6,8,6,6,
     3,8,3,5,6,5,3,8,3,0,0
1830 REM Rock 2
1840 DATA 5,6,5,6,6,6,6,3,5,6,5,6,5,3,6,
     3,6,3,6,6
1850 DATA 5,6,5,6,6,6,6,3,5,6,5,6,5,3,1,
     3,1,3,2,3,2,3,0,0
1860 REM Rock 3
1870 DATA 5,6,8,6,6,3,8,6,6,3,5,6,6,3,8,
     3,5,3,6,3,8,3,5,3,0,0
1880 REM Mexican Rock
1890 DATA 10,18,6,12,6,6,5,6,5,6
1900 DATA 10,18,6,12,1,3,1,3,2,3,2,3,4,3
     ,4,3,0,0
1910 REM Insertion 1
1920 DATA 8,6,8,6,6,6,8,6,8,6,6,3,6,3,1,
     3,1,3,4,3,4,3,0,0
1930 REM Insertion 2
1940 DATA 1,4,1,4,1,4,2,4,2,4,2,4,3,4,3,
     4,3,4,4,4,4,4,4,4,0,0
1950 REM Insertion 3
1960 DATA 1,3,1,3,1,3,1,3,2,3,2,3,2,3,2,
     3,3,3,3,3,3,3,3,3,4,3,4,3,4,3,4,3,0
     ,0
1970 REM Swing
1980 DATA 5,12,7,8,7,4,7,12,7,8,7,4
1990 DATA 7,12,7,8,7,4,7,12,7,8,4,4,0,0
2000 REM Cha Cha
2010 DATA 11,12,11,6,5,6,11,12,11,12
2020 DATA 11,12,11,6,5,6,11,6,11,6,11,6,
     5,6,0,0
```

Le rythme du départ du programme est celui du rock. En suivant les indications, vous pourrez changer de rythme.

Commentaires

Chaque élément de percussion est produit par la combinaison des paramètres: hauteur, enveloppe de volume, enveloppe de tonalité et bruitage qui sont récupérés dans quatre variables **array** en ligne 1520.

Les rythmes sont listés dans des instructions DATA, en fin de programme. Ils sont stockés sous forme de numéro de percussion suivi d'une durée. Le numéro de chaque élément de percussion est donné dans les instructions REM de la ligne 1550 à la ligne 1760. Les variables de rythme sont toutes terminées par deux zéros.

La boucle principale WHILE/WEND entre les lignes 1080 et 1120 procède à la lecture du numéro de percussion et de sa durée et produit le son correspondant. Si une nouvelle touche est appuyée, la ligne 1090 procède au stockage dans **temps$;** cette variable est utilisée de 1130 à 1210 pour ramener le bon jeu de paramètres.

A la fin, le programme passe par 1120 puis tombe sur une série de lignes identiques, de 1130 à 1210. En effet, l'AMSTRAD ne supporte pas l'instruction "RESTORE line" où line est une variable contenant un résultat, celui d'un calcul par exemple. Sinon nous aurions été en mesure de faire un RESTORE à une ligne calculée d'avance et nous aurions économisé ainsi un peu de place mémoire.

Le bruit des percussions n'est pas exclu dans la composition de musiques électroniques. Tous les bruits n'ont pu être simulés dans notre exemple de programme. Faites évoluer votre exemple en changeant les paramètres et essayez de connecter votre AMSTRAD sur des enceintes

externes pour voir l'effet produit. Vous pouvez également changer la mesure ou le tempo en agisssant sur les paramètres de la ligne 1780.

Comment utiliser ces effets dans vos programmes utilitaires

Il est évident que l'explosion produite par un rayon laser sera considérée comme inopportune en plein milieu d'un programme de gestion. Mais par contre d'autres effets peuvent rendre vos programmes plus intéressants et en faciliter la mise en oeuvre.

Beaucoup de personnes, dont moi-même, n'aiment pas être rappelées à l'ordre par des bips sonores en cas d'erreur. En tout cas, l'AMSTRAD utilise le bip suivant:

```
PRINT CHR$(7)
```

La facilité d'utilisation d'un programme est déterminée par son architecture, ses affichages et surtout par ses routines de surveillance des entrées au clavier. Nous n'examinerons pas ces points en détail car ils sont en dehors du contexte de ce livre. Mais dans un programme avec affichage des différentes options il est facile d'inclure une petite routine rappelant à l'opérateur une mauvaise sélection. Quelque chose du genre:

```
SOUND 1,0,10,5,0,0,2
```

L'opérateur sera tout de suite informé s'il a appuyé sur la bonne touche ou non. A la place d'un vulgaire bip de rappel à l'ordre, vous pouvez utiliser un son plus doux. Les erreurs sont surtout dues à la mauvaise interprétation des instructions et rien n'est plus frustrant que de recevoir un bip en retour de ses efforts. Il faudrait plutôt renvoyer à une autre partie du programme ou donner à l'opérateur de nouvelles directives.

Si un programme traverse plusieurs sections, il est possible également d'incorporer une petite mélodie ou fanfare, comme dans le test de rapidité du chapitre 1, pour indiquer qu'une partie du programme a été réalisée. De telles fanfares peuvent être construites à partir de l'enveloppe de tonalité.

Les suggestions ci-dessus rendront votre programme de travail plus intéressant et plus vivant. Il serait tout de même sage de prévoir une option qui supprime éventuellement le son, en sorte de ne pas imposer votre création musicale, aussi merveilleuse soit-elle.

L'ambiance maritime: un programme complet d'effets sonores

Avec un ordinateur comme l'AMSTRAD associé à un générateur de son, vous pouvez créer des effets sonores de trains, de bateaux, de voitures, de jungle... Les plus intéressants sont ceux ayant une période de répétition très longue ou qui n'ont pas de répétition du tout. Le programme suivant utilise les principes et les idées que nous venons d'exposer:

```
100 REM PROGRAM 7.3
110 REM Mer, Ecume & Mouettes
120 :
130 ENV 1,1,15,3,15,-1,8:REM Mouette
140 ENT 1,20,1,2:REM Mouette
150 ENV 2,5,3,1,2,0,175,15,-1,3:REM Cor
    ne de brume
160 :
170 EVERY 750 GOSUB 410:REM Corne de bru
    me
180 ON SQ(1) GOSUB 270:REM Vagues
190 :
200 WHILE -1
210 REM Mouettes
220 IF INT(RND*300+1)=1 THEN SOUND 2,INT
    (RND*15+1)+15,42,0,1,1
230 IF INT(RND*300+1)=1 THEN SOUND 4,INT
    (RND*30+1)+30,42,0,1,1
240 WEND
250 :
```

```
260 REM Vagues
270 déferlement=INT(RND*18+1)+2
280 étalement=INT(RND*200+1)
290 retrait=INT(RND*90+1)+10
300 vague=INT(RND*25+1)+6
310 calme1=INT(RND*9+1)
320 calme2=INT(RND*4+1)*50
330 ENV 3,15,1,déferlement

340 ENV 4,1,0,étalement,13,-1,retrait,
    calme 1,0,calme2
350 SOUND 1,0,0,0,3,0,vague
360 SOUND 1,0,0,15,4,0,vague-6
370 ON SQ(1) GOSUB 270
380 RETURN
390 :
400 REM Corne de brume
410 SOUND 162,1073,400,0,2
420 SOUND 148,1074,400,0,2
430 RETURN
```

Commentaires

La plus grande partie du programme est dévolue à la génération de nombres aléatoires dans une gamme pouvant nous donner des résultats intéressants. Les lignes 130 à 150 mettent en place les enveloppes pour les bruits de mouettes et de corne de brume.

En 170, une interruption logicielle est placée pour mettre la corne de brume en route toutes les quinze secondes. La routine de la corne de brume utilise les canaux B et C. La ligne 410 vide les sons dans le canal B avec un rendez-vous sur le canal C. 420 est identique mais envoie les sons dans le canal B. Les vidages sont nécessaires pour émettre en même temps le bruit des mouettes et le bruit de la corne de brume. Variez la hauteur des notes pour produire un bruit de trompe différent.

Une interruption pour le générateur de son SQ(1) est mise en place en ligne 180 pour appeler la routine simulant le bruit des vagues en 270. SQ a été expliqué au chapitre 3.

L'interruption est répétée en ligne 370. Cette routine redéfinit périodiquement les enveloppes 3 et 4 pour produire des sons de vagues. Chaque vague utilise deux hauteurs de bruit et les effets générés par les enveloppes de volume vont du calme plat à la tempête.

La boucle principale du programme entre les lignes 200 et 240 lance le bruit des mouettes en utilisant une variation de l'enveloppe de ricochet du programme 7.1. Si les cris vous paraissent trop électroniques, diminuez la gamme des hauteurs. Vous pouvez également changer l'image stéréo en jouant sur les affectations des canaux.

Comment aller plus loin

Bien que nous ne disposions pas des mêmes possibilités qu'avec des synthétiseurs plus puissants, nous pouvons malgré tout produire des effets de fond utilisables dans un programme. Un programme d'études graphiques pourrait être facilement greffé sur notre exemple de paysage maritime, l'AMSTRAD pouvant produire une gamme très étendue d'effets graphiques et sonores.

8

Jouer sur l'Amstrad

L'introduction des générateurs de sons dans les ordinateurs familiaux ou personnels a donné naissance à un nouvel instrument de musique: l'ordinateur. La facilité d'utilisation de cet instrument dépend de sa composition matérielle et de la complexité des sons logiciels.

L'Amstrad, un instrument de musique

Dans la plupart des instruments de musique l'accent est mis sur l'ergonomie pour qu'on puisse en jouer facilement dans les limites imposées par la forme de l'instrument. Le meilleur exemple est le piano pour lequel il existe bien d'autres formes de claviers supposés plus aisés à pratiquer.

Les ordinateurs n'ayant pas été spécialement étudiés pour être utilisés comme instruments ne sont pas équipés d'un clavier musical et si nous voulons interpréter un morceau, il faudra le faire au travers du clavier QWERTY.

Selon votre faculté d'adaptation, vous trouverez cela plus ou moins difficile. Le clavier QWERTY n'a aucun rapport avec un clavier de piano, les touches sont différentes et les débutants peuvent avoir un certain avantage sur les pianistes.

Avec un peu de pratique, vous acquerrez des automatismes pour certaines séquences de notes.

Il en est de même pour les dactylos sur les claviers QWERTY, quand elles créent certaines suites de mots. Que vous soyez musicien ou non, vous constaterez qu'un peu de pratique améliorera votre utilisation du clavier QWERTY comme instrument de musique, mais il vous sera difficile de devenir virtuose.

Instruments monophoniques, instruments polyphoniques

Un instrument monophonique est un instrument ne pouvant générer qu'une seule note à la fois. La plupart des instruments font partie de cette catégorie: flûte, trompette, violon... Bien que techniquement la plupart d'entre eux puissent jouer plus d'une note, ils sont tout de même catalogués comme monophoniques. D'autres instruments comme le piano, l'orgue, la harpe peuvent émettre plusieurs notes à la fois et sont classés polyphoniques.

Un synthétiseur peut être décrit comme monophonique ou polyphonique. Souvent les synthétiseurs classés monophoniques sont en réalité des duophoniques et peuvent émettre deux notes à la fois. La catégorie des polyphoniques émet de quatre à seize notes. Avec l'accroissement de l'intégration des circuits électroniques et leur coût de plus en plus faible, nous trouverons une majorité de synthétiseurs polyphoniques sur le marché.

L'Amstrad, en tant qu'instrument polyphonique

Il y a plusieurs façons d'écrire un programme nous permettant de jouer de la musique à partir d'un clavier. Le programme suivant illustre l'une de ces manières:

```
100 REM PROGRAM 8.1
110 REM Clavier monophonique
120 REM De F#-1 à D#1
130 :
140 ON BREAK GOSUB 270
150 SPEED KEY 255,255
```

```
160 clavier$="1q2w3er5t6yu8i9o0p@^[
    (16)REM Dernière note=Touche CLR
170 ENV 1,1,15,1,2,-1,2,13,-1,10
180 :
190 WHILE -1
200 jouer$=INKEY$:IF jouer$="" THEN 200
210 note=INSTR(clavier$,jouer$)-6
220 freq=440*(2^(0+(note-10)/12))
230 hauteur=ROUND(125000/freq)
240 IF hauteur<677 THEN SOUND 129,
    hauteur,0,0,1
250 WEND
260 :
270 SPEED KEY 20,3
280 STOP
```

Commentaires

Le programme est si simple que nous aurions presque pu le résumer en deux ou trois lignes. La ligne 150 diminue la fréquence de répétition des touches et la 140 repositionne la vitesse initiale à la suite d'un ESC. **clavier$** contient un certain nombre de touches que nous allons utiliser, chaque touche de gauche à droite incrémente la fréquence d'un demi-ton. Si vous vous référez à la figure 2.4, elles correspondent aux notes: FA -1 à RE 1.

Les entrées sont détectées par INKEY$, transférées dans **jouer$** et utilisées dans la fonction INSTR pour produire le numéro de la note. On retranche 6 à la note en ligne 210, ainsi une touche 1 jouera en FA -1. Il est possible d'utiliser d'autres valeurs que 6 pour faire démarrer le clavier à une autre note.

La fréquence est déduite à partir de la note, puis la hauteur. Si la hauteur est dans les limites requises, un son est émis. Par contre si nous appuyons sur une touche non reconnue par **clavier$**, la note sera affectée de la valeur -1, la hauteur sera égale à 716 et aucun son ne pourra être émis. La commande SOUND contient un paramètre de vidage pour sortir la note immédiatement.

L'Amstrad, en tant que synthétiseur

A notre modeste programme nous pouvons ajouter quelques routines pour produire des effets de synthétiseur:

```
1000 REM PROGRAM 8.2
1010 REM Synthétiseur Amstrad
1020 REM De F#-1 à D#1
1030 :
1040 ON BREAK GOSUB 1900
1050 SPEED KEY 255,255
1060 ENV 1,1,15,1,2,-1,2,13,-1,10
1070 ENV 2,16,15,2
1080 ENV 3,15,1,1,4,-1,10,11,-1,25
1090 ENV 4,1,15,17,8,7,17,1,-7,17,1,5,17
     ,2,-2,17
1100 ENT -1,1,1,2,1,-2,2,1,1,2
1110 ENT -2,2,1,2,4,-1,2,2,1,2
1120 ENT 3,1,100,1,10,-10,1
1130 clavier$="1q2w3er5t6yu8i9o0p@^["+
     CHR$(16):dernière note=touche CLR
1140 oct=0:ev=1:et=0
1150 GOSUB 1630:REM Dessin du clavier
1160 GOSUB 1560:REM Impression des
     flèches de l'enveloppe
1170 :
1180 WHILE -1
1190 jouer$=INKEY$:IF jouer$="" THEN
     1190
1200 IF jouer$=CHR$(240) OR jouer$=CHR$
     (241)THEN GOSUB 1360:GOTO 1190:REM
     Octave
1210 REM note-6 fait que "1" entendu
     F#-1
1220 note=INSTR(clavier$,jouer$)-6
1230 IF note>-6 THEN GOSUB 1290:GOTO
     1190:REM Jouer une note
1240 IF jouer$=" " THEN GOSUB 1430:GOTO
     1190:REM Liaison des chorus par
     touche TAB
1250 GOSUB 1480:REM Altération des Envel
     loppes
1260 WEND
1270 :
1280 REM Jouer une note
```

```
1290 freq=440*(2^(oct+(note-10)/12))
1300 hauteur=ROUND(125000/freq)
1310 SOUND 129,hauteur,0,0,ev,et
1320 IF chorus THEN SOUND 130,hauteur-1,
     0,0,ev,et
1330 RETURN
1340 :
1350 REM Changement d'octave
1360 IF jouer$=CHR$(240) THEN oct=oct+1
     ELSE oct=oct-1
1370 IF oct<-2 THEN oct=3
1380 IF oct>3 THEN oct=-2
1390 LOCATE 38,18:PRINT oct;
1400 RETURN
1410 :
1420 REM Chorus
1430 chorus=NOT chorus
1440 LOCATE 29,16:PRINT ABS(chorus);
1450 RETURN
1460 :
1470 REM Altération des Enveloppes
1480 IF jouer$="z" THEN ev=1
1490 IF jouer$="x" THEN ev=2
1500 IF jouer$="c" THEN ev=3
1510 IF jouer$="v" THEN ev=4
1520 IF jouer$="a" THEN et=0
1530 IF jouer$="s" THEN et=1
1540 IF jouer$="d" THEN et=2
1550 IF jouer$="f" THEN et=3
1560 LOCATE 1,21:PRINT SPC(19)
1570 LOCATE 4+5*et,21:PRINT CHR$(240)
1580 LOCATE 1,22:PRINT SPC(19)
1590 LOCATE 4+5*(ev-1),22:PRINT CHR$
     (241)
1600 RETURN
1610 :
1620 REM Dessin de clavier
1630 MODE 1
1640 INK 0,13:INK 1,0
1650 k1$=" "+CHR$(143)+CHR$(143)
1660 k2$=k1$+k1$
1670 k3$=k2$+k1$+"  "+CHR$(211)
1680 k4$=k3$+k2$+"  "+CHR$(211)+k3$+k2$
1690 k5$="  "+CHR$(211):FOR n=1 TO 13:
     k6$=k6$+k5$:NEXT
1700 PRINT:PRINT " 1 2 3   5 6   8 9  0
          ^"
```

```
1710 FOR n=1 TO 5:PRINT k4$:NEXT
1720 FOR n=1 TO 6:PRINT k6$:NEXT
1730 PRINT CHR$(11)+CHR$(22)+CHR$(1);"";
1740 FOR n=1 TO 36:PRINT CHR$(210);:NEXT
1750 PRINT CHR$(11);CHR$(22);"   Q  W  E
     R  T  Y  U  I  O  P  @  [";
1760 PRINT CHR$(22)+CHR$(0)
1770 :
1780 PRINT:PRINT TAB(35);CHR$(240)
1790 PRINT TAB(34);"oct:";oct;
1800 PRINT TAB (35);CHR$(241)
1810 :
1820 LOCATE 18,16:PRINT "chorus TAB: 0"
1830 :
1840 LOCATE 1,19:PRINT "  A   S   D   F"
1850 PRINT " ENT0 ENT1 ENT2 ENT3"
1860 PRINT:PRINT:PRINT "   Z   X   C   V"
1870 PRINT " ENV1 ENV2 ENV3 ENV4"
1880 RETURN
1890 :
1900 ON BREAK STOP
1910 SPEED KEY 20,3
1920 STOP
```

Au départ, notre programme dessine un clavier de piano sur l'écran avec les touches QWERTY correspondantes. L'octave est changée avec les touches de positionnement du curseur, les flèches pointant vers le haut et le bas. La touche TAB met en route ou non l'effet de chorus; A,S,D et F vous permettent de choisir les enveloppes de tonalité, et Z,X,C et V les enveloppes de volume.

Commentaires

Le programme commence par diminuer la vitesse de répétition des touches et par définir les enveloppes que nous allons utiliser. clavier$ est identique au dernier exemple mais maintenant nous pouvons changer d'octave et d'enveloppe et les valeurs par défaut sont mises en place en ligne 1140.

En 1150, nous faisons appel à la routine 1630 qui dessine le clavier. Le dessin est fait

à partir de six variables caractères, construites en ligne 1650 à 1690 et dessinées de 1700 à 1760. Les lignes suivantes affichent les contrôles ainsi que les lettres à utiliser pour les touches.

La routine en 1560, qui est appelée par 1160, permet d'afficher la flèche de l'enveloppe. En 1160 nous nous connectons sur la routine en 1560. Cette routine est la dernière des routines de sélection des enveloppes et elle affiche le pointeur de l'enveloppe à la bonne place. La boucle principale est située entre 1180 et 1260. En 1190 nous attendons les entrées au clavier. Si une touche curseur est appuyée, nous appellerons une routine de changement d'octave. Par contre si la touche est comprise dans **clavier$** la note sera supérieure à -6 et la ligne 1230 se connectera sur la routine de génération du son. Si TAB est appuyé, la ligne 1240 nous débranchera sur la routine d'effet de chorus. Les autres touches changent les enveloppes et nous en tiendrons compte dans les routines d'altération des enveloppes en 1480.

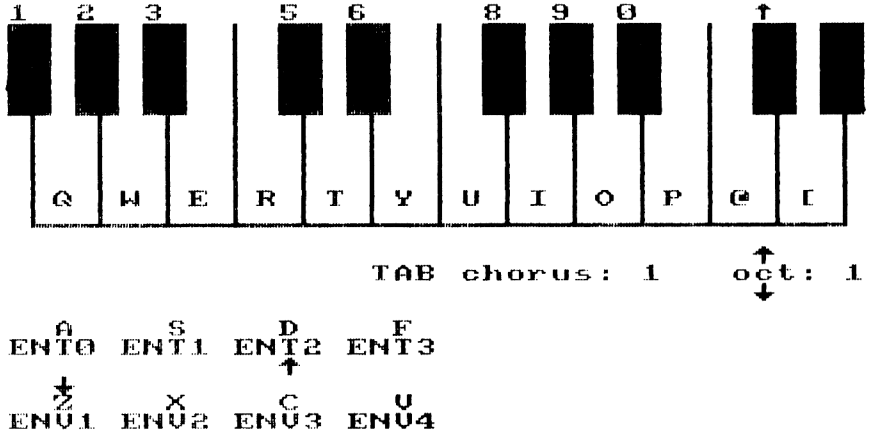


Figure 8.1. Sortie écran du programme 8.2.

Quelques explications sont encore nécessaires. De 1290 à 1300, nous calculons la fréquence puis la hauteur comme d'habitude, et

en 1320 l'effet de chorus. De 1360 à 1390, nous surveillons les changements d'octaves car il y a recouvrement d'octaves si vous essayez de sortir de la gamme.

La routine d'effet de chorus bascule la variable chorus entre 0 (off) et -1 (on). Néanmoins, nous afficherons 1 en 1440 à la place de -1 pour faire plus net. De 1480 à 1550 nous changerons les enveloppes. Les lignes 1570 et 1590 se chargent du calcul des pointeurs pour choisir la bonne enveloppe.

Affichage du clavier

Le fait d'avoir un diagramme des différentes touches avec un marquage des boutons ou des touches caractéristiques peut être d'une grande utilité lorsque nous abordons des instruments qui ne nous sont pas familiers. Il est possible également de coller de petits morceaux de papier, des caches, sur les touches non utilisées.

COMMENT CHANGER OU DEVELOPPER LE PROGRAMME

Les effets sont tous très courts et la plupart des idées et des principes ont été abordés dans les chapitres précédents. On peut changer les touches de contrôle pour les rendre plus commodes pour un gaucher.

Ajoutez plus d'effets, plus d'enveloppes également, même toutes les quinze si vous le désirez. Vous pouvez mêler à tout cela un effet de bruitage. Votre synthétiseur fera plus "pro" si vous ajoutez une touche qui interprétera vos notes un octave en dessous pour étoffer le son.

Si vous voulez passer pour un virtuose du clavier, vous trouverez peut-être le temps de réponse de l'AMSTRAD un peu long. En fait ce petit délai n'est pas un problème et vous pouvez le minimiser en comprimant vos instructions, en utilisant des variables à un seule lettre et en

regroupant plusieurs instructions sur une même ligne.

Ajout d'un séquenceur de basses

Le séquenceur est un ensemble programmable utilisé dans les synthétiseurs pour produire un jeu de notes de manière répétitive. Il varie dans sa sophistication, le nombre de notes programmables pouvant aller de douze à quelques milliers de notes.

De nombreux synthétiseurs du commerce contrôlent l'émission des notes par l'amplitude du signal, ainsi plus l'amplitude est élevée, plus la hauteur de note est élevée. De la même manière, quelques séquenceurs stockent leurs notes sous forme d'une liste de tensions. Dans les synthétiseurs pilotés en amplitude, vous trouverez les modules suivants: l'oscillateur ou VCO (Voltage Controlled Oscillator), l'amplificateur ou VCA (Voltage Controlled Amplifier), le filtre ou VCF (Voltage Controlled Filter).

A chaque utilisation d'une instruction DATA pour lire une chaîne de notes, nous utilisons, en fait, un séquenceur.

Nous pouvons ajouter un séquenceur de basses à notre dernier programme pour nous jouer un rythme sur lequel nous pourrions improviser. Ceci est assez commun en musique électronique et certains musiciens comme Tangerine Dream, Kraftwerk ou Jean-Michel Jarre ont popularisé ces effets. Le prochain listing nous montre les lignes à ajouter au programme 8.2

```
10 REM PROGRAM 8.3
20 REM Séquenceur de basses
30 REM Ajouter ces lignes au PROGRAM 8.2
40 :
50 tempo=4
60 ENV 15,1,15,1,1,-2,2,13,-1,5
70 ENT -5,2,3,1,2,-6,1,2,3,1
```

```
  80 ON SQ(4) GOSUB 2010
  90:
2000 REM Routine du séquenceur
2010 READ pit,dur
2020 IF pit=0 THEN RESTORE 2070:GOTO
     2010
2030 SOUND 4,pit,dur*tempo,0,15,15
2040 ON SQ(4) GOSUB 2010
2050 RETURN
2060 :
2070 DATA 804,6,758,6,956,6,1073,12,1073
     ,12,1136,6,1073,12,1073,6,1136,6,
     1073,6,1136,6,1073,6,956,6
2080 DATA 804,6,758,6,956,6,1073,12,1073
     ,12,1136,6,1073,12,1073,6,1136,6,
     1073,24,0,0
2090 DATA 638,6,536,6,402,6,426,6,0,0
2100 DATA 638,24,536,24,568,24,602,24,0,0
     0
2110 DATA 638,6,638,6,536,6,478,6,0,0
```

Commentaires

La séquence utilise ENV 15 ainsi qu'ENT 15 qui sont définis dans les lignes de 60 à 70. La routine principale, de 2010 à 2050, est de toute simplicité, elle ne fait que lire une note et la reproduire. Elle utilise une interruption ON SEQ GOSUB mise en place en 80 et réarmée en 2040. Le programme tournera par lui-même si vous lui ajoutez:

```
85 WHILE -1:WEND
```

La séquence des données se termine par un double zéro.

Pour changer l'accompagnement

Une séquence supplémentaire a été incluse entre les lignes 2090 et 2110 pour votre expérimentation, une autre pouvant facilement être ajoutée. Essayez la séquence en ligne 2100 avec une enveloppe plus longue:

```
ENV 15,1,15,1,1,-2,10,13,-1,15
```

Changez l'enveloppe de tonalité pour qu'elle s'adapte.

Les thèmes d'accompagnement dans le programme sont très simples mais ils peuvent être étoffés en augmentant le nombre de données. Vous pouvez également programmer votre séquenceur pour qu'il joue une partition complète de basse.

Comment améliorer le séquenceur

Vous pouvez ajouter un rythme à votre séquenceur en prenant comme base les idées développées dans le programme 7.2 et cela devrait être intéressant.

En dehors du fait d'ajouter plus d'enveloppes, plus de rythmes, plus de séquences et plus d'effets, vous allez peut-être ajouter une mémoire à votre programme pour garder trace de tout ce que vous avez improvisé et le rejouer par la suite. Ceci pourra vous être très utile: ne vous est-il jamais arrivé de faire une improvisation terrible puis d'être incapable de la refaire dans la minute qui suit?

Pour réaliser cela, il est nécessaire d'inclure une variable incrémentée et d'enregistrer sa valeur dès qu'une touche est appuyée ou relâchée. Les notes seront stockées dans un array.

Votre séquenceur passera au stade supérieur si vous lui ajoutez une possibilité d'édition, avec une option pour changer les paramètres (hauteur, tempo, enveloppe,etc.). Vous atteignez alors le stade de la console d'enregistrement miniature. De plus, ceci constituera sûrement un exercice de programmation très utile.

Si vous voulez contrôler et changer la musique à un tel degré, l'usage du clavier QWERTY n'arrange en rien vos affaires.

Peut-être est-il préférable d'enregistrer la pièce entière et de laisser l'AMSTRAD se débrouiller dans les passages difficiles. C'est ce que nous examinerons dans le prochain chapitre.

9

Composer de la musique sur micro-ordinateur

Après avoir essayé les programmes du chapitre 8, vous avez sûrement réalisé que le clavier QWERTY n'est pas un instrument des plus facile à maîtriser. Il est très bien pour improviser sur un rythme produit par un séquenceur, mais dès qu'il s'agit d'interpréter une mélodie en temps réel, l'affaire se complique. Il est encore plus difficile de réaliser des accords de deux ou trois notes sur ce type de clavier.

Derrière le clavier nous avons un ordinateur et nous devrions pouvoir surmonter ce problème assez facilement en lui fournissant tous les ingrédients nécessaires: hauteur, enveloppe, durée... lui laissant la charge de combiner tout cela. L'ordinateur peut interpréter des séquences que nous ne saurions jamais réaliser sur le clavier, il le fait très bien et à n'importe quel instant.

La méthode la plus simple et encore la meilleure consiste à passer les données dans un programme au travers des instructions DATA. Cette méthode est utilisée dans la plupart des programmes de ce livre. Ceci convient surtout à des interprétations très courtes mais dès que nous dépassons les trente à quarante articles la méthode se complique sérieusement. Il nous faut trouver autre chose, une méthode qui nous permettrait d'entrer les notes sous une autre forme qu'une suite de nombres. Examinons quelques options.

Parlons musique

La notation conventionnelle nous permet de communiquer nos idées musicales à toute personne parlant le même langage. Comme cette notation est mondialement connue et acceptée, nous n'avons pratiquement aucun problème d'interprétation, sauf pour les subtilités artistiques qui sont difficiles à transcrire. A ce jour, les micro-ordinateurs n'ont pas encore été éduqués en matière de musique, ce qui nous laisse le libre choix de la méthode pour communiquer avec eux. Ceci peut être très intéressant étant donné qu'il n'y a aucune standardisation dans le domaine des micros. La communication que nous allons établir ne pourra être utilisée que par nous-mêmes. Etant donné toutes ces considérations, nous ne nous inquiéterons pas outre mesure de la manière de discuter musicalement avec l'AMSTRAD et nous aborderons plusieurs solutions.

Il est toujours intéressant de voir comment les autres constructeurs de micro-ordinateurs ont résolu le problème. Quelques-uns passent par la méthode du patchwork en utilisant l'instruction POKE pour remplir des positions mémoire. D'autres sont plus proches de l'AMSTRAD en utilisant des commandes SOUND ou des instructions PLAY et quelques-uns permettent même à l'utilisateur de spécifier une note particulière par un nom ou un numéro d'octave. Très peu d'ordinateurs peuvent synchroniser trois canaux, et ceux qui le peuvent utilisent généralement le code machine.

Ce que nous devons retenir à ce stade, c'est qu'aucun vendeur de micro n'a conçu à ce jour un bon programme Basic pour l'interprétation musicale. Il faut dire que le Basic n'est pas le langage idéal pour la programmation musicale, aussi quelques concepteurs de micro sont-ils passés au langage MCL (Music Composition Language) : le prix de leurs micro-ordinateurs ne se situe pas dans la même gamme que celui de l'AMSTRAD. Néanmoins, des logiciels musicaux

pour ordinateurs personnels font leur apparition sur le marché et quelques-uns sont livrés avec un langage permettant de franchir le fossé qui sépare le musicien de l'ordinateur. A ce jour aucun n'a été accepté comme standard.

Nous sommes donc complètement libres de mettre en place un système qui sied parfaitement à notre ordinateur. Nos besoins peuvent varier et nous pouvons étudier plusieurs systèmes adaptés à nos différentes applications. Nous avons insisté tout au long de ce livre sur le fait que nous nous tiendrions à la notation conventionnelle. Il y a plusieurs méthodes pour transcrire les notes en nombres. Nous allons analyser ces méthodes mais auparavant il convient de faire une remarque à l'attention du lecteur français.

Remarque sur les notations anglaise et française

Ce livre est traduit de l'anglais et nous arrivons maintenant au stade où il est impossible de passer sous silence les problèmes de conversion entre la notation anglaise et la notation française. Rassurez-vous, cela n'est pas bien compliqué: au lieu d'appeler les notes DO, RE, MI..., en anglais on se contente de leur donner une lettre de l'alphabet:

DO RE MI FA SOL LA SI - notation française

C D E F G A B - notation anglaise

Ainsi un F 3 correspond à un FA de la troisième octave. Il est nécessaire de s'habituer aux deux notations et nous introduirons la notation anglaise au fur et à mesure. En effet, les programmes permettant de faire la correspondance entre une notation conventionnelle et un nombre ont tous été réalisés à partir de la notation anglaise. Pour adapter les programmes à la représentation française, il aurait fallu retranscrire tous les programmes.

Le marché de l'informatique étant pratiquement entre les mains des Anglo-Saxons il peut sembler nécessaire de savoir transcrire un SOL en G et A en LA. Au début on donnera les deux représentations pour faciliter la transcription.

La méthode numérique

C'est la méthode utilisée par la commande SOUND, où une liste de nombres correspond à une fréquence ou plutôt une hauteur particulière, par exemple 478 correspond au C ou DO du milieu du clavier. Ceci est parfait, mais dès que vous voulez rentrer une grande suite de notes, il est difficile de s'y retrouver. Les ordinateurs sont friands de nombres, mais les humains préfèrent généralement des choses un peu moins abstraites.

Dès qu'il s'agit d'une petite mélodie, il faut choisir cette méthode. Si nous ne dépassons pas quelques douzaines de notes, nous n'aurons pas d'ennuis sérieux pour entrer et éditer les notes; d'ailleurs c'est la méthode que nous avons utilisée jusqu'à présent.

Les autres méthodes

L'idée de départ est de convertir en nombre des notes entrées sous forme compréhensible. Il existe plusieurs variantes, les différences étant situées sur la méthode de codage des notes et des préférences personnelles.

Il existe plusieurs formes de notation pour spécifier une note dans une portée. La plus utilisée est celle qui consiste à faire suivre le nom de la note par un chiffre représentant le numéro de l'octave: DO1 ou C1, B5 ou SI5, F 3 ou FA 3 etc. D'autres systèmes emploient un jeu d'apostrophes pour indiquer les octaves: DO'(C'), LA''(A''), RE ''' etc. ou RE''' , SOL'' . Quelques systèmes placent des lignes horizontales, directement au-dessus de la note, d'autres des chiffres romains pour indiquer

l'octave, il y en a même qui comptent en demi-tons de telle manière que DO1 sera suivi de DO 2,RE3,RE 4, etc.

La version plus facile à comprendre, celle qui produit le moins d'erreurs, est la version avec numérotation des octaves. Si vous n'êtes pas convaincu, libre à vous de changer de méthode pour l'adapter à vos propres besoins. De toute manière, le principe de représentation de l'octave est toujours le même, la seule chose qui change est la façon de le représenter.

Dans l'annexe 2 du guide d'utilisation il est d'ailleurs fait référence au même système et le DO se trouvant au milieu du clavier du piano est numéroté C0. D'autres systèmes similaires mettent le numéro d'octave en indice ou en exposant.

Une alternative avec le système de numérotation des octaves

Dans ce livre, à l'annexe 1, dans la figure 2.4, et en plus de la représentation du guide d'utilisation, vous pouvez voir un autre jeu de numérotation. Notre préférence irait plutôt vers cette forme de représentation pour deux raisons. Premièrement, parce que la longueur maximale d'une note est de trois caractères, sauf pour les octaves négatives où la longueur passe à quatre comme : C -1. Deuxièmement, parce que la routine de conversion sera d'autant plus réduite. Nous en retirerons deux avantages : l'espace-mémoire nécessaire sera plus réduit, et l'exécution sera plus rapide.

Cependant, étant donné que beaucoup de personnes ont appris la programmation à travers le guide d'utilisation, elles sont plus familiarisées avec son système de représentation des octaves; le mieux est donc de se conformer à cette méthode. Si d'autres programmeurs suivent cet exemple, il y aura sûrement une plus grande compatibilité entre les programmes de musique sur l'AMSTRAD.

Le système de notation décrit ci-dessous peut toujours être adapté à une autre notation si le besoin s'en fait sentir. Le chapitre 4 liste un certain nombre de formules pour des systèmes de numérotation différents.

La donnée est entrée comme un nom de note plus un numéro d'octave et une durée. Le fait de stocker ainsi la note nous permet de visualiser immédiatement l'information et de la traiter ultérieurement.

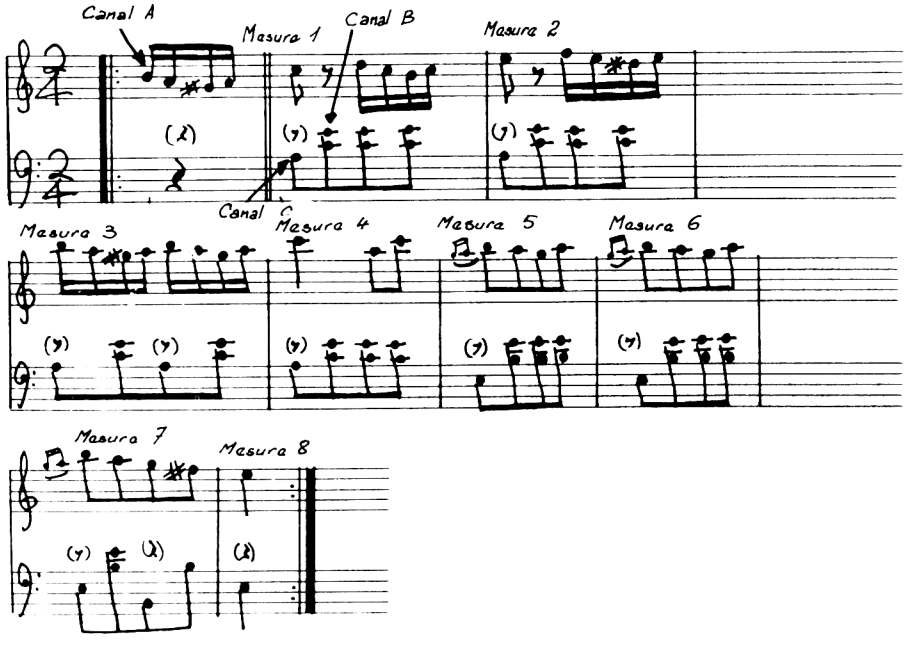


Figure 9.1. Les silences entre parenthèses sont pris en compte et se réfèrent à la chaîne.

La durée des notes

La durée d'une note est normalement un multiple de la plus petite note de la pièce. Dans l'exemple de la figure 9.1, la note la plus petite est une double croche à laquelle il faudra donner une valeur. En fait, nous lui donnerons la valeur 3. Ceci nous permettra

d'avoir une base de calcul pour déterminer la valeur des autres notes. Si une double croche a une valeur de 3, une noire aura une durée de 12. Pour faire correspondre trois notes (triolet) dans le temps d'une noire, il faudra leur donner une valeur 4. Regardez le programme 7.2 et les données de Fill 2 et Fill 3 comme exemple. Dans un regroupement identique de trois croches, la valeur est ramenée à 2.

Un rythme standard sur les synthétiseurs est de 24 ppqn (24 "pulses per quarter note") ou en bon français 24 impulsions par noire. Avec une telle base, la valeur de notre double croche est 6 (1 noire vaut 4 doubles croches).

Ceci constitue simplement un guide, vous êtes libre d'utiliser la durée qui s'adapte le mieux à votre musique.

Un programme de conversion note/nombre

Il est basé sur les illustrations du guide d'utilisation, annexe 1, figure 2.4.

```
100 REM PROGRAM 9.1
110 REM Rondo turc de Mozart
120 REM 1 seul canal
130 :
140 gamme$=" C C#D D#E F F#G G#A A#B"
150 tempo=4
160 ev1=1
170 ENV 1,1,15,1,3,-1,4,12,-1,8
180 :
190 FOR n=1 TO 46
200 READ note$,dur
210 GOSUB 270:REM Analyse de Note
220 SOUND 1,hauteur,dur*tempo,0,ev
230 NEXT n
240 END
250 :
260 REM Analyse de note
270 IF note$="R" THEN ev=0:RETURN ELSE
    ev=ev1
280 longueur=LEN(note$):IF longueur<2 OR
```

```
    longueur>4 THEN PRINT "erreur note";
    n;note$:STOP
290 IF longueur=2 THEN nom$=LEFT$(note$,
    1):oct=VAL(RIGHT$(note$,1)):GOTO 320
300 IF longueur=4 THEN nom$=LEFT$(note$,
    2):oct=VAL(RIGHT$(note$,2)):GOTO 320
310 IF MID$(note$,2,1)="-" THEN nom$=
    LEFT$(note$,1):oct=VAL(RIGHT$(note$,
    2)) ELSE nom$=LEFT$(note$,2):oct=VAL
    (RIGHT$(note$,1))
320 note=INSTR(gamme$,nom$)/2
330 freq=440*(2^(oct+(note-10)/12))
340 hauteur=ROUND(125000/freq)
350 RETURN
360 :
370 DATA B0,3,A0,3,G#0,3,A0,3
380 DATA C1,6,R,6,D1,3,C1,3,B0,3,C1,3
390 DATA E1,6,R,6,F1,3,E1,3,D#1,3,E1,3
400 DATA B1,3,A1,3,G#1,3,A1,3,B1,3,A1,3,
    G#1,3,A1,3
410 DATA C2,12,A1,6,C2,4,G1,1,A1,1
420 DATA B1,6,A1,6,G1,6,A1,4,G1,1,A1,1
430 DATA B1,6,A1,6,G1,6,A1,4,G1,1,A1,1
440 DATA B1,6,A1,6,G1,6,F#1,6
450 DATA E1,12
```

Ce programme inteprète les huit premières mesures du Rondo Turc de Mozart. Il existe une version sur trois canaux comme nous le montre la figure 9.1, mais nous la programmerons un peu plus tard.

Commentaires

La variable **gamme$** est définie en ligne 140 et contient toutes les notes de la gamme, les espaces ou blancs sont importants, les noires sont toutes converties en dièses. Généralement, ce n'est pas un problème pour nous de définir les notes de cette manière, mais si le morceau comprend des bémols, nous sommes obligés de faire une conversion mentale en dièse. La figure 2.4 peut nous aider. De plus, le clavier comporte le sigle uniquement, ce qui nous évite de faire des confusions entre la lettre b et le symbole bémol.

La boucle principale entre les lignes 190 et 230 étant du type lecture et émission de note, nous avons juste un appel à la routine 270 pour analyser la note.

Vous avez remarqué les deux petites notes au début des mesures 5 et 6. Nous n'allons pas nous replonger dans une grande théorie musicale, nous indiquerons seulement qu'il faut jouer ces notes très rapidement avant la "vraie" note de la mesure. Ce sont des notes d'"appoggiature"; avec la base de temps que nous avons définie tout à l'heure, nous leur donnerons une valeur de 1.

Si **note$** est un R, nous avons affaire à un silence ou REST en anglais, et ev est mis à 0. Sinon la note est analysée.

La variable **note$** peut avoir une longueur de 2,3 ou 4 caractères. Aucune vérification n'est faite sur les caractères dièses ou bémols. Si **note$** a une longueur de 2 caractères, le premier représente automatiquement le nom de la note, notation anglaise SVP, le second le numéro de l'octave. Si la longueur est de 4, la note est définie sur les deux premiers caractères, par exemple F et le numéro d'octave dans les deux autres. Ce contrôle est effectué en 300.

Si la longueur de **note$** est de 3, il existe deux possibilités: soit une note avec un dièse ou un bémol dans une octave positive ou une note sans accident dans une octave négative. L'analyse est faite en ligne 310.

Le nombre affecté à la note est extrait de **gamme$** en 320, puis on détermine la fréquence et la hauteur comme toujours. Au retour de la routine, la note est jouée.

UTILISATION D'UN SYSTEME DIFFERENT DE NUMEROTATION DES OCTAVES

Si notre numérotation d'octave est uniquement positive, la routine d'analyse pourra être

simplifiée:

```
IF LEN(note$)=2 THEN nom$=LEFT$(note$,1) ELSE
nom$=LEFT$(note$,2) oct=VAL(RIGHT$(note$,1))
```

Nous pouvons alors récrire la formule de fréquence en ligne 330 ou ajuster **oct** en soustrayant 4 dans l'exemple ci-dessus.

Les routines d'erreurs

Il est toujours intéressant d'inclure des routines d'erreurs qui nous indiquent en temps réel une mauvaise entrée d'un paramètre. Un tel contrôle est incorporé dans la ligne 280. Les données sont arrangées en lignes d'une mesure à chaque fois, pour faciliter l'entrée, la lecture et le déverminage.

Emission de mélodies composées

Il est très facile de jouer des mélodies de deux ou trois airs si les notes dans les différents canaux ont la même durée; une simple modification dans le dernier programme est alors suffisante. Un problème se pose quand les notes des différents canaux ont des durées différentes.

Ce cas peut être mis en évidence avec la figure 9.1. Vous pouvez voir que pendant les trois premières mesures, vingt-quatre notes auront passé dans le canal A, et seulement treize dans les autres canaux. La solution de mettre les notes dans une file d'attente ne nous avance pas beaucoup. La solution réside dans le stockage des notes dans une variable array d'où nous pouvons extraire les notes au moment voulu. L'utilisation des commandes ON SQ et des paramètres de rendez-vous vous faciliteront la tâche.

La stéréo

Normalement les canaux sont arrangés de la façon suivante:

Canal A : Mélodie
Canal C : Basse
Canal B : Tout ce qu'il faut pour renforcer la basse/accompagnement

Il arrive que, dans des morceaux où la basse est plus importante, on la programme en canal A.

Tout ceci nous amène à prendre en compte le positionnement stéréo des voix. Le phénomène stéréo est surtout sensible si vous connectez votre AMSTRAD sur des enceintes externes ou si vous branchez un casque. Généralement la basse est dirigée sur une voie, la mélodie sur l'autre et tout ce qui se trouve entre les deux sur le canal B qui en assure une distribution centrale. Les trois voix sont stockées indépendamment et vous pouvez les diriger sur n'importe quel canal. Faites l'expérience de diriger les voix sur différents canaux.

Le caractère d'une pièce est donné par la mélodie et la basse, ce sont donc eux qu'il faut veiller à placer correctement. On utilise le canal B pour l'harmonie, et si nécessaire on bascule de la voix basse sur la voix mélodie. Avec cet arrangement les canaux A et C seront servis avant le canal B, ce qui peut sembler bizarre au travers du listing.

Le programme suivant nous donne le contrôle de cinq paramètres sur chaque note. En pratique, si nous tenons compte des quelques considérations que nous avons édictées dans les chapitres précédents, il est possible de simplifier le programme. Il n'y a que de rares cas de figure où nous aurons besoin des cinq paramètres pour définir une note.

```
1000 REM PROGRAM 9.2
1010 REM Rondo turc de Mozart
1020 REM 3 canaux utilisés
1030 :
1040 va=46:vb=29:vc=30
1050 :
1060 REM première subscription se réfère à:
1070 REM 1=statut
1080 REM 2=hauteur
1090 REM 3=dur
1100 REM 4=ENV
1110 REM 5=ENT
1120 :
1130 DIM canala(5,va)
1140 DIM canalb(5,vb)
1150 DIM canalc(5,vc)
1160 :
1170 gamme$=" C C#D D#E F F#G G#A A#B "
1180 tempo=4
1190 ENV 1,1,15,1,3,-1,4,12,-1,8
1200 ENT 1
1210 ENV 2,3,5,1,1,0,8,1,-2,4,13,-1,8
1220 ENT -2,1,1,3,1,-2,3,1,1,3
1230 ENV 3,1,15,1,3,-1,2,12,-1,6
1240 ENT 3
1250 :
1260 CLS:PRINT "Lecture des notes..."
1270 FOR n=1 TO va
1280 GOSUB 1740:REM Vérification statut
     canal
1290 Canala(1,n)=statut+1
1300 GOSUB 1780:REM Analyse de note
1310 IF note$="R" THEN ev=0 ELSE IF n=5
     OR n=11 OR n=25 OR n=30 OR n=36 OR
     n=42 THEN ev=2 ELSE ev=1
1320 et=ev
1330 canala(2,n)=hauteur
1340 canala(3,n)=dur
1350 canala(4,n)=ev
1360 canala(5,n)=et
1370 NEXT n
1380 PRINT "Canal A complet"
1390 :
1400 FOR n=1 TO vc
1410 GOSUB 1740:REM Vérification statut
     canal
```

```
1420 canalc(1,n)=statut+4
1430 GOSUB 1780:REM Analyse de note
1440 IF note$="R" THEN ev=0 ELSE ev=3
1450 et=ev
1460 canalc(2,n)=hauteur
1470 canalc(3,n)=dur
1480 canalc(4,n)=ev
1490 canalc(5,n)=et
1500 NEXT n
1510 PRINT "Canal C complet"
1520 :
1530 FOR n=1 TO vb
1540 GOSUB 1740:REM Vérification statut
     canal
1550 canalb(1,n)=statut+2
1560 GOSUB 1780:REM Analyse de note
1570 IF note$="R" THEN ev=0 ELSE ev=1
1580 et=ev
1590 canalb(2,n)=hauteur
1600 canalb(3,n)=dur
1610 canalb(4,n)=ev
1620 canalb(5,n)=et
1630 NEXT n
1640 PRINT "Cannal B complet"
1650 :
1660 voca=0:vocb=0:vocc=0
1670 ON SQ(1) GOSUB 1890
1680 ON SQ(2) GOSUB 1950
1690 ON SQ(4) GOSUB 2010
1700 WHILE voca<va OR vocb<vb OR vocc<vc
     :WEND
1710 END
1720 :
1730 REM Vérification statut canal
1740 READ note$:IF LEFT$(note$,1)="&"
     THEN statut=VAL(note$):READ note$,
     dur ELSE statut=0:read DUR
1750 RETURN
1760 :
1770 REM Analyse de note
1780 IF note$="R" THEN hauteur=0:RETURN
1790 longueur=LEN(note$):IF longueur<2
     OR longueur> 4 THEN PRINT "Erreur
     note";n;note$:STOP
1800 IF longueur=2 THEN nom$=LEFT$(note$
     ,1):oct=VAL(RIGHT$(note$,1)):GOTO
     1830
```

```
1810 IF longueur=4 THEN nom$=LEFT$(note$
     ,2):oct=VAL(RIGHT$(note$,2)):GOTO
     1830
1820 IF MID$(note$,2,1)="-" THEN nom$=
     LEFT$(note$,1):oct=VAL(RIGHT$(note$
     ,2)) ELSE nom$=LEFT$(note$,2):oct=
     VAL(RIGHT$(note$,1))
1830 note=INSTR(gamme$,nom$)/2
1840 freq=440*(2^(oct+(note-10)/12))
1850 hauteur=ROUND(125000/freq)
1860 RETURN
1870 :
1880 REM Canal A
1890 voca=voca+1:IF voca>va THEN RETURN
1900 SOUND canala(1,voca),canala(2,voca)
     ,canala(3,voca)*tempo,0,canala(4,
     voca),canala(5,voca)
1910 ON SQ(1) GOSUB 1890
1920 RETURN
1930 :
1940 REM Canal B
1950 vocb=vocb+1:IF vocb>vb THEN RETURN
1960 SOUND canalb(1,vocb),canalb(2,vocb)
     ,canalb(3,vocb)*tempo,0,canalb(4,
     vocb),canalb(5,vocb)
1970 ON SQ(2) GOSUB 1950
1980 RETURN
1990 :
2000 REM Canal C
2010 vocc=vocc+1:IF vocc>vc THEN RETURN
2020 SOUND canalc(1,vocc),canalc(2,vocc)
     ,canalc(3,vocc)*tempo,0,canalc(4,
     vocc),canalc(5,vocc)
2030 ON SQ(4) GOSUB 2010
2040 RETURN
2050 :
2060 REM Voix 1
2070 DATA &38,B0,3,A0,3,G#0,3,A0,3
2080 DATA C1,6,R,6,D1,3,C1,3,B0,3,C1,3
2090 DATA E1,6,R,6,F1,3,E1,3,D#1,3,E1,3
2100 DATA B1,3,A1,3,G#1,3,A1,3,B1,3,A1,3
     ,G#1,3,A1,3
2110 DATA C2,12,A1,6,C2,4,G1,1,A1,1
2120 DATA B1,6,A1,6,G1,6,A1,4,G1,1,A1,1
2130 DATA B1,6,A1,6,G1,6,A1,4,G1,1,A1,1
2140 DATA B1,6,A1,6,G1,6,F#1,6
```

```
2150 DATA E1,12
2160 :
2170 REM Voix 2
2180 DATA &38,R,12
2190 DATA A-1,6,C0,6,C0,6,C0,6
2200 DATA A-1,6,C0,6,C0,6,C0,6
2210 DATA A-1,6,C0,6,A-1,6,C0,6
2220 DATA A-1,6,C0,6,C0,6,C0,6
2230 DATA E-1,6,B-1,6,B-1,6,B-1,6
2240 DATA E-1,6,B-1,6,B-1,6,B-1,6
2250 DATA E-1,6,B-1,6,B-1,6,B-1,6
2260 DATA E-1,12
2270 :
2280 REM Voix 3
2290 DATA &38,R,12
2300 DATA R,6,E0,6,E0,6,E0,6
2310 DATA R,6,E0,6,E0,6,E0,6
2320 DATA R,6,E0,6,R,6,E0,6
2330 DATA R,6,E0,6,E0,6,E0,6
2340 DATA R,6,E0,6,E0,6,E0,6
2350 DATA R,6,E0,6,E0,6,E0,6
2360 DATA R,6,E0,6,R,12
2370 DATA R,12
```

Commentaires

Pour réaliser l'affectation des variables à chaque canal, nous faisons suivre d'une lettre. Les désignations utilisées sont 1,2 ou 3 ou alors A,B ou C. La désignation par les chiffres semble plus naturelle, mais si nous voulons nous référer aux canaux, il vaut mieux utiliser la désignation en A,B,C.

Les variables: va, vb et vc contiennent les notes de chaque voix. Trois variables **array** ont été mises en place entre 1130 et 1150 pour pouvoir contenir les informations sur chaque note. Les indices sont définis dans le tableau suivant:

```
indice = 1: Statut, numéro du canal,
            rendez-vous, vidage...
indice = 2: Hauteur
indice = 3: Durée
```

indice = 4: ENV
indice = 5: ENT

La mesure est mise en place et les enveloppes sont définies. Le programme effectue la liaison entre l'enveloppe de volume et l'enveloppe de tonalité. Il faudra prendre l'habitude de remettre à zéro toutes les enveloppes à la fin d'un programme. Il se peut bien que votre AMSTRAD utilise une des deux enveloppes d'un programme précédent à un moment ou à un autre. La section suivante met en place les données dans les variables **array** correspondantes et ceci pour chaque canal.Il est possible d'éviter ces répétitions en regroupant toutes les données dans une seule variable **array** à trois dimensions, du type (x,y,z), mais la compréhension n'en serait que plus complexe. Le déroulement étant exactement le même pour les trois canaux, nous n'examinerons en détail que le canal A.

L'écran est remis à blanc et un message est imprimé pour nous laisser voir que l'ordinateur travaille. En effet, pour certaines interprétations, l'AMSTRAD peut mettre de une à deux minutes pour digérer le programme et il est réconfortant de voir arriver un message.

Une boucle FOR/NEXT balaye les données. Le premier pas est un appel à la routine en 1740 et l'on examine avec beaucoup de circonspection la première donnée. Si elle commence avec un & elle sait que nous avons affaire à une instruction concernant le canal et elle passe ensuite à la lecture de **note$** et de **dur.** Le statut du canal est sous forme hexadécimale et le signe & nous indique que nous sommes en présence d'une valeur d'état du canal. Vous pouvez utiliser la notation décimale en ignorant l'&. Consultez l'annexe 2 dans le guide d'utilisation pour avoir plus de détails sur la représentation hexadécimale.

Si les différentes données ne commencent pas par un &, il est reconnu que nous sommes en

présence de notes et une seconde lecture est effectuée dans **dur**. La variable **statut** est mise à 0. Le fait de n'assigner une valeur de statut qu'en cas de besoin est assez pratique. En 1290, nous assignons le statut au canal et nous ajoutons 1, qui est, en fait, le numéro du canal.

La routine en 1780 est appelée pour analyser la note. Cette routine est identique à celle du programme 9.1 avec juste une petite modification. Si la **note$** est un silence, une valeur arbitraire de 0 est donnée à la hauteur. Cela n'est absolument pas obligatoire mais est très utile dans la phase de déverminage.

Par la suite, **note$ est examinée pour** rechercher un R qui représente un silence. Si ce R est trouvé, l'enveloppe de volume est mise à zéro. Ceci a pour effet de produire un son dont le volume est déterminé par le paramètre V de la commande SOUND que nous avons préalablement mis à 0. Ainsi aucun son ne pourra être émis. Sinon ev **est programmé à la** valeur du numéro d'enveloppe requis. Reportez-vous aux lignes 1440 et 1570 où une seule autre enveloppe est utilisée.

1310 est arrangée pour faire correspondre à certaines notes certaines enveloppes. Cela nous fait gagner du temps. L'enveloppe de tonalité et **est réglée égale à l'enveloppe de volume** ev en 1320. Une routine similaire à celle de 1310 peut être utilisée pour faire varier l'enveloppe de tonalité.

Les lignes suivantes copient toutes ces informations dans les variables array correspondantes. Le message de la 1380 nous informe que les données du canal ont été stockées.

A la fin de ces trois sections, les variables array **sont remplies d'informations** directement exploitables par une commande SOUND, mais elles peuvent encore être modifiées comme nous allons le voir.

La ligne 1660 positionne trois variables que nous utiliserons pour vérifier le nombre de notes envoyées à chaque canal. Les trois lignes qui suivent mettent en place les interruptions logicielles ON SQ. Dès que les premières notes arrivent au rendez-vous, l'émission commence en synchro. Pendant le temps que le générateur de sons émet ses premières notes, l'ordinateur aura pu traiter quelques instructions et remplir la file d'attente. L'émission des premières notes aurait également pu se faire par un HOLD sur le canal, un remplissage de la file d'attente, puis une commande RELEASE pour libérer la musique.

En 1890, 1950 et 2010 nous trouvons trois routines similaires. A chaque passage dans la routine, nous incrémentons la variable voc de 1. Si la valeur est supérieure au nombre de notes dans le canal, nous sortons de la routine. La ligne suivante émet simplement une commande SOUND. A ce stade nous contrôlons également la vitesse d'exécution de la pièce avec tempo.

La boucle WHILE/WEND en 1700 ne fait rien de plus que de garder le programme en route pendant que les interruptions font le travail. Si vous le désirez, vous pouvez changer le rythme en faisant la modification suivante:

```
GOTO 1660
```

Suggestions et modifications

Maintenant que vous avez ce programme, vous pouvez insérer de nouvelles données et changer quelques variables. Si vous programmez d'autres tonalités et que tous les canaux ne démarrent pas ensemble, assurez-vous d'inclure des temps de pause dans les canaux inutilisés pour les garder en phase. Vous avez pu remarquer que seules les premières notes sont vraiment synchrones. Théoriquement cela devrait suffire pour jouer correctement, mais vous pouvez ajouter plusieurs autres commandes de rendez-vous si nécessaire. L'instruction

d'interruption ON SQ permet au Basic de réaliser d'autres commandes pendant l'émission des sons. Dans ce programme, la boucle en 1700 est inutile. Si le programme est en train de réaliser d'autres fonctions, il est peut être nécessaire d'inclure d'autres rendez-vous.

Les changements d'enveloppes sont toujours ennuyeux, et s'ils sont nombreux il vous est conseillé d'écrire un petite routine à part. Dans le calcul des durées il est important de veiller à ce que la relation entre notes reste correcte. L'ajustement de la variable tempo peut être nécessaire également.

Déverminage des données

Ce serait vraiment exceptionnel si vous arriviez à rentrer toutes les données sans une seule erreur. La routine d'analyse détecte une simple erreur de longueur. Vous allez être informé du canal où s'est glissée l'erreur puisqu'un message est affiché à chaque fois qu'un canal est rempli correctement avec des notes.

Une erreur que l'on commet souvent est d'inverser les valeurs de va, vb et vc,si bien que chaque canal se remplit avec les notes du canal voisin. Vous pouvez parfaire votre détection d'erreur en insérant un caractère de fin dès que toutes les données d'un canal sont entrées. Par exemple, si note$ réceptionne un "§" et que cette valeur ne correspond pas à va vous allez vous poser des questions. Cette méthode est également utilisable pour détecter le nombre de données et dimensionner correctement les array. Rappelez-vous également que va=46 en début de programme et qu'après la boucle FOR/NEXT en 1370, va sera égal à 47.

Si toutes les données sont correctes, nous pouvons supprimer les routines de détection des erreurs.

La sauvegarde des informations

Vous pouvez utiliser le programme de contrôle des données puis sauvegarder les informations sur cassette ou sur disque. Ceci signifie que les données n'ont pas à être stockées deux fois, dans les instructions DATA et dans les array, et qu'il n'y aura pas de routine d'analyse de note. On met plus de temps pour charger les cassettes que le programme original n'en prend pour analyser la mélodie, si bien que nous ne gagnerons pas forcément du temps. Le programme principal doit contenir les caractéritiques d'ENV et ENT ainsi que la routine permettant de jouer les notes. Cette dernière peut être condensée, ce qui nous permettra de gagner de la place mémoire. Vous pouvez vous constituer une librairie de mélodies sous forme de fichiers et écouter vos mélodies quand l'envie vous en prend. Normalement le programme principal s'occupe des enveloppes, mais vous pouvez ajouter une routine qui sauvegardera ces paramètres.

Un fichier peut être sauvegardé tout simplement en insérant les lignes suivantes dans le programme 9.2

```
   10 REM PROGRAM 9.3
   20 REM Sauvegarde des airs dans fichiers
   30 REM Insérer ces lignes
   40 REM Dans PROGRAM 9.2
 1710 :
 1712 OPENOUT "RONDO"
 1713 WRITE #9,tempo,va,vb,vc
 1714 FOR n=1 TO va:FOR m=1 TO 5
 1715 WRITE #9,canala(m,n)
 1716 NEXT m:NEXT n
 1717 FOR n=1 TO vb:FOR m=1 TO 5
 1718 WRITE #9,canalb(m,n)
 1719 NEXT m:NEXT n
 1720 FOR n=1 TO vc:FOR m=1 TO 5
 1721 WRITE #9,canalc(m,n)
 1722 NEXT m:NEXT n
 1723 CLOSEOUT
 1724 END
 1725 :
```

Commentaires

Si vous êtes satisfait de l'arrangement d'une mélodie, faites tourner le programme. Dès que l'air est fini entrez: GOTO 1172. En ligne 1712 on procède à l'ouverture d'un fichier pour bande appelé RONDO. Le tempo et les notes de chaque canal sont sauvegardés, suivis par le contenu des array. **Le fichier est fermé en** 1723.

Le programme destiné au chargement et à l'exécution de la mélodie est dérivé du programme 9.2

```
1000 REM PROGRAM 9.4
1010 REM Chargement des données d'un
     fichier
1020 REM et exécution du morceau
1030 :
1040 REM Définition des enveloppes
1050 ENV 1,1,15,1,3,-1,4,12,-1,8
1060 ENT 1
1070 ENV 2,3,5,1,1,0,8,1,-2,4,13,-1,8
1080 ENT -2,1,1,3,1,-2,3,1,1,3
1090 ENV 3,1,15,1,3,-1,2,12,-1,6
1100 ENT 3
1110 :
1120 CLS:PRINT "Lecture des notes..."
1130 OPENIN "RONDO"
1140 INPUT #9,tempo,va,vb,vc
1150 :
1160 DIM canala(5,va)
1170 DIM canalb(5,vb)
1180 DIM canalc(5,vc)
1190 :
1200 FOR n=1 TO va:FOR m=1 TO 5
1210 INPUT #9,canala(m,n)
1220 NEXT m:NEXT n
1230 FOR n=1 TO vb:FOR m=1 TO 5
1240 INPUT #9,canalb(m,n)
1250 NEXT m:NEXT n
1260 FOR n=1 TO vc:FOR m=1 TO 5
1270 INPUT #9,canalc(m,n)
1280 NEXT m:NEXT n
1290 CLOSEIN
```

```
1300 PRINT:PRINT "Voici l'air..."
1310 :
1320 voca=0:vocb=0:vocc=0
1330 ON SQ(1) GOSUB 1400
1340 ON SQ(2) GOSUB 1460
1350 ON SQ(4) GOSUB 1520
1360 WHILE voca<va OR vocb<vb OR vocc<vc
     :WEND
1370 END
1380 :
1390 REM Canal A
1400 voca=voca+1:IF voca>va THEN RETURN
1410 SOUND canala(1,voca),canala(2,voca)
     ,canala(3,voca)*tempo,0,canala(4,
     voca),canala(5,voca)
1420 ON SQ(1) GOSUB 1400
1430 RETURN
1440 :
1450 REM Canal B
1460 vocb=vocb+1:IF vocb>vb THEN RETURN
1470 SOUND canalb(1,vocb),canalb(2,vocb)
     canalb(3,vocb)*tempo,0,canalb(4,
     vocb),canalb(5,vocb)
1480 ON SQ(2) GOSUB 1460
1490 RETURN
1500 :
1510 REM Canal C
1520 vocc=vocc+1:IF vocc>vc THEN RETURN
1530 SOUND canalc(1,vocc),canalc(2,vocc)
     canalc(3,vocc)*tempo,0,canalc(4,
     vocc),canalc(5,vocc)
1540 ON SQ(4) GOSUB 1520
1550 RETURN
```

Si les enveloppes ne sont pas définies, le programme utilisera celles laissées en place par le programme précédent. La ligne 1130 ouvre le fichier et les lignes suivantes effectuent les opérations de lecture.

Expériences avec les programmes

Le programme 9.2 n'est qu'un début que vous pouvez améliorer. Si la quinzaine d'enveloppes n'est pas suffisante, vous pouvez toujours procéder à une redéfinition en milieu de

programme, dans la boucle WHILE/WEND par exemple.

Il est possible également de superposer un effet de bruit sur un canal, ou même un fond rythmique. Vous pouvez également produire votre propre arrangement sur vos airs préférés et nous verrons cela au prochain chapitre. Mais avant...

VOICI D'AUTRES MELODIES

Ces trois programmes contiennent les données et les informations pour jouer d'autres airs. Chargez le programme 9.2 et entrez les lignes suivantes sans changer quoi que ce soit. Quelques lignes seront blanches (avec juste un :), ce qui signifie qu'elles ont été effacées du programme original.

Le programme 9.5 joue les seize mesures suivantes du Rondo turc, avec une répétition des huit premières.

```
  10 REM PROGRAM 9.5
  20 REM Encore du Mozart
  30 REM Rondo turc
  40 REM Insérer ces lignes
  50 REM dans PROGRAM 9.2
  60 :
1040 va=168:vb=114:vc=114
1245 ENV 4,1,13,1,1,0,30,13,-1,6
1246 ENT -4,1,-14,6,1,14,6
1275 IF n=47 OR n=129 THEN RESTORE 2070
1276 IF n=153 THEN RESTORE 2159
1310 IF note$="R" THEN ev=0 ELSE IF n=51
     OR n=57 OR n=71 OR n=76 OR n=82 OR
     n=88 THEN ev=2 ELSE ev=1
1315 IF n=153 OR n=165 THEN ev=4
1405 IF n=31 OR n=85 THEN RESTORE 2180
1406 IF n=98 THEN RESTORE 2269
1535 IF n=30 OR n=85 THEN RESTORE 2290
1536 IF n=98 THEN RESTORE 2379
2060 REM Voix 1
```

```
2150 DATA E1,12,E1,6,F1,6
2151 DATA G1,6,G1,6,A1,3,G1,3,F1,3,E1,3
2152 DATA D1,12,E1,6,F1,6
2153 DATA G1,6,G1,6,A1,3,G1,3,F1,3,E1,3
2154 DATA D1,12,C1,6,D1,6
2155 DATA E1,6,E1,6,F1,3,E1,3,D1,3,C1,3
2156 DATA B0,12,C1,6,D1,6
2157 DATA E1,6,E1,6,F1,3,E1,3,D1,3,C1,3
2158 DATA B0,12
2159 DATA C2,12,A1,6,B1,6
2160 DATA C2,6,B1,6,A1,6,G#1,6
2161 DATA A1,6,E1,6,F1,6,D1,6
2162 DATA C1,12,B0,10,A0,1,B0,1
2163 DATA A0,12
2170 REM Voix 2
2260 DATA E-1,12,R,12
2261 DATA C-1,6,C0,6,E-1,6,E0,6
2262 DATA G-1,12,R,12
2263 DATA C-1,6,C0,6,E-1,6,E0,6
2264 DATA G-1,12,R,12
2265 DATA C-1,6,A-1,6,C-1,6,C0,6
2266 DATA E-1,12,R,12
2267 DATA C-1,6,A-1,6,C-1,6,C0,6
2268 DATA E-1,12
2269 DATA F-1,6,A-1,6,A-1,6,A-1,6
2270 DATA E-1,6,A-1,6,D-1,6,F-1,6
2271 DATA C-1,6,E-1,6,D-1,6,F-1,6
2272 DATA E-1,6,E-1,6,E-1,6,E-1,6
2273 DATA A-1,12
2280 REM Voix 3
2370 DATA R,12,C1,6,D1,6
2371 DATA E1,6,E1,6,R,12
2372 DATA B0,6,G0,6,C1,6,D1,6
2373 DATA E1,6,E1,6,R,12
2374 DATA B0,12,A0,6,B0,6
2375 DATA C1,6,C1,6,R,12
2376 DATA G#0,6,E0,6,A0,6,B0,6
2377 DATA C1,6,C1,6,R,12
2378 DATA R,12
2379 DATA R,6,D#0,6,D#0,6,D#0,6
2380 DATA R,6,E0,6,R,6,B-1,6
2381 DATA R,6,A-1,6,R,6,B-1,6
2382 DATA A-1,6,A-1,6,G#-1,6,G#-1,6
2383 DATA A-2,12
```

Remarquez comme les répétitions ont été faciles à programmer avec l'instruction RESTORE. Les seules autres additions sont ENV4 et ENT4 pour le trille. Le premier trille , en DO2 est décalé d'une valeur de 1 hauteur, l'autre en SI0 est parfait.

```
  10 REM PROGRAM 9.6
  20 REM John Philip Sousa
  30 REM Marche de la Liberté
  40 REM Insérer ces Lignes
  50 REM dans PROGRAM 9.2
  60 :
1040 va=174:vb=179:vc=209
1180 tempo=5
1190 ENV 1,7,2,1,14,-1,10
1210 ENV 2,7,2,2,4,-1,4,10,-1,6
1230 ENV 3,1,13,4,13,-1,30
1240 ENT -3,1,0,4,1,-9,4,1,9,4
1245 ENV 4,5,3,1,4,-1,6,11,-1,8
1246 ENT -4,1,4,2,1,-8,2,1,4,2
1275 IF n=65 THEN RESTORE 2110
1310 IF note$="R" THEN ev=0 ELSE IF (n=1
     OR n=78 OR n=93 OR n=71 OR n=108 OR
     n=162) THEN ev=3 ELSE ev=1
1405 IF n=80 THEN RESTORE 2500
1440 IF note$="R" THEN ev=0 ELSE IF
     n=208 THEN ev=1 ELSE IF (n>79 AND
     n<144) THEN ev=4 ELSE ev=2
1535 IF n=74 THEN RESTORE 2890
1570 IF note$="R" THEN ev=0 ELSE IF (n=
     133 THEN ev=3 OR ELSE IF n=178 THEN
     ev=1 ELSE ev=2
2060 REM Voix 1
2070 DATA &38,F2,63,R,6
2080 :
2090 :
2100 DATA C1,3
2110 DATA &38,A0,6,A0,3,A0,3,G#0,3,A0,3
2120 DATA F1,6,C1,3,C1,6,A0,3
2130 DATA A#0,6,A#0,3,A#0,6,C1,3
2140 DATA D1,15,A#0,3
2150 DATA G0,6,G0,3,G0,3,F#0,3,G0,3
2160 DATA E1,6,D1,3,D1,6,A#0,3
2170 DATA A0,6,A0,3,A0,6,A#0,3
2180 DATA C1,15,C1,3
```

```
2190 DATA A0,6,A0,3,A0,3,G#0,3,A0,3
2200 DATA A1,6,F1,3,F1,6,C1,3
2210 DATA B0,6,G1,3,G1,6,G1,3
2220 DATA G1,15,F1,3
2230 DATA E1,6,G1,3,G1,3,F#1,3,G1,3
2240 DATA D1,6,G1,3,G1,3,F#1,3,G1,3
2250 DATA C1,6,B0,3,C1,6,B0,3
2260 DATA C1,9,C1,9
2270 REM Deuxième partie
2280 DATA A0,3,G#0,3,A0,3,D1,6,C1,3
2290 DATA A0,9,F0,9
2300 DATA D0,9,G0,9
2310 DATA F0,15,F0,3
2320 DATA G0,3,A0,3,A#0,3,E1,6,D1,3
2330 DATA C1,9,F1,9
2340 DATA E1,9,D1,9
2350 DATA C1,15,C1,3
2360 DATA D1,6,D1,2,E1,1,D1,3,C1,3,D1,3
2370 DATA E1,9,E1,9
2380 DATA F1,6,F1,2,A1,1,G1,3,F1,3,G1,3
2390 DATA A1,15,A1,2,A1,1
2400 DATA G1,6,F1,3,D1,6,A#0,3
2410 DATA A0,9,F0,9
2420 DATA G0,9,E0,9
2430 DATA F0,12,R,6
2440 :
2450 REM Voix 2
2460 DATA &38,F0,6,E0,3,D#0,6,D0,3
2470 DATA C0,6,B-1,3,A#-1,6,A-1,3
2480 DATA G-1,3,A-1,3,A#-1,3,A-1,6,G-1,3
2490 DATA C-1,6,R,12
2500 DATA &38,F-1,6,F0,3,F0,6,F0,3
2510 DATA F-1,6,F0,3,F0,6,F0,3
2520 DATA E-1,6,F-1,3,G-1,6,F-1,3
2530 DATA E-1,6,D-1,3,C-1,6,R,3
2540 DATA C-1,6,A#-1,3,A#-1,6,A#-1,3
2550 DATA C-1,3,A#-1,3,A#-1,6,A#-1,3
2560 DATA F-1,6,G-1,3,A-1,6,G-1,3
2570 DATA F-1,6,E-1,3,D-1,6,C-1,3
2580 DATA F-1,6,F0,3,F0,6,F0,3
2590 DATA F-1,6,F0,3,F0,6,F0,3
2600 DATA D-1,6,E-1,3,F-1,6,E-1,3
2610 DATA D-1,6,C-1,3,B-2,6,A-1,3
2620 DATA G-1,6,E0,3,E0,6,E0,3
2630 DATA G-1,6,F0,3,F0,6,F0,3
2640 DATA C0,6,G-1,3,C0,6,G-1,3
```

```
2650 DATA C-1,6,R,3,C0,6,R,3
2660 REM Deuxième partie
2670 DATA F-1,6,A-1,3,C-1,6,A-1,3
2680 DATA F-1,6,A-1,3,C-1,6,A-1,3
2690 DATA A#-1,6,F-1,3,C-1,6,E-1,3
2700 DATA F-1,6,A-1,3,A-1,6,A-1,3
2710 DATA E-1,6,A#-1,3,C-1,6,A#-1,3
2720 DATA F-1,6,A-1,3,D-1,6,G#-1,3
2730 DATA C-1,4,F-1,2,G-1,3,D-1,4,F-1,2,
     G-1,3
2740 DATA C-1,6,G-1,3,G-1,6,G-1,3
2750 DATA B-2,6,B-1,3,B-1,6,B-1,3
2760 DATA C#-1,6,A-1,3,A-1,6,A-1,3
2770 DATA D-1,6,A-1,3,D-1,6,A#-1,3
2780 DATA C-1,6,A-1,3,A-1,6,A-1,3
2790 DATA G-1,6,A#-1,3,A#-1,6,A#-1,3
2800 DATA C-1,6,A-1,3,A-1,6,A-1,3
2810 DATA E-1,6,A#-1,3,C-1,6,A#-1,3
2820 DATA F-1,6,R,3,F-1,6,R,3
2830 :
2840 REM Voix 3
2850 DATA &38,F1,6,E1,3,D#1,6,D1,3
2860 DATA C1,6,B0,3,A#0,6,A0,3
2870 DATA G0,3,A0,3,A#0,3,A0,6,G0,3
2880 DATA C0,6,R,12
2890 DATA &38,F0,6,C0,3,C0,6,C0,3
2900 DATA F0,6,C0,3,C0,6,C0,3
2910 DATA G0,6,G0,3,G0,6,G0,3
2920 DATA G0,15,R,3
2930 DATA G-1,6,C0,3,C0,6,C0,3
2940 DATA G-1,6,C0,3,C0,6,C0,3
2950 DATA F0,6,F0,3,F0,6,G0,3
2960 DATA A0,15,A0,3
2970 DATA F0,6,C0,3,C0,6,C0,3
2980 DATA F0,6,C0,3,C0,6,C0,3
2990 DATA D0,9,B-1,9
3000 DATA B-1,6,C0,3,D0,6,R,3
3010 DATA G0,6,C0,3,C0,6,C0,3
3020 DATA G0,6,B-1,3,B-1,6,B-1,3
3030 DATA E0,6,R,3,G-1,6,R,3
3040 DATA E0,6,R,3,E0,6,R,3
3050 REM Deuxième partie
3060 DATA R,18
3070 DATA F2,54
3080 :
3090 :
```

```
3100 DATA R,6,C0,3,R,6,C0,3
3110 DATA R,6,A-1,3,R,6,G#-1,3
3120 DATA G0,9,F0,9
3130 DATA R,6,C0,3,C0,6,C0,3
3140 DATA F0,6,D0,3,D0,6,D0,3
3150 DATA R,6,C#0,3,C#0,6,C#0,3
3160 DATA R,6,D0,3,R,6,D0,3
3170 DATA R,6,E0,3,E0,6,E0,3
3180 DATA R,6,G0,3,G0,6,G0,3
3190 DATA R,6,C0,3,C0,6,C0,3
3200 DATA R,6,C0,3,C0,6,C0,3
3210 DATA F0,6,R,3,F0,6,R,3
```

L'instruction RESTORE est utilisée pour les les répétitions et plusieurs variations de ton. La lenteur de l'attaque des enveloppes imite bien les cuivres, néanmoins il vaut mieux essayer plusieurs enveloppes. Peut-être voudrez-vous définir des enveloppes de tonalité individuelles pour chaque trille.

```
  10 REM PROGRAM 9.7
  20 REM Tschaïkowsky
  30 REM Danse de la Fée Prune
  40 REM Insérer ces lignes
  50 REM dans PROGRAM 9.2
  60 :
1040 va=104:vb=86:vc=95
1180 tempo=10
1190 ENV 1,2,7,1,14,-1,8
1210 ENV 2,1,12,1,12,-1,8
1220 ENT -2,1,1,3,2,-1,3,1,1,3
1230 ENV 3,1,10,1,10,-1,10
1245 ENV 4,1,12,1,12,-1,5
1246 ENT -4,1,1,5,2,-1,5,1,1,5
1247 ENV 5,1,12,1,12,-1,8
1248 ENT -5,1,2,2,1,-4,2,1,2,2
1310 IF note$="R" THEN ev=0 ELSE IF n=va
     THEN ev=5 ELSE IF n<17 THEN ev=2
     ELSE ev=1
1440 IF note$="R" THEN ev=0 ELSE IF n=vc
     THEN ev=5 ELSE ev=2
1570 IF note$="R" THEN ev=0 ELSE IF n=vb
     THEN ev=5 ELSE IF (n>36 AND n<48)
     THEN ev=3 ELSE ev=2
2060 REM Voix 1
```

```
2070 DATA &38,R,4,E1,4,R,4,F#1,4
2080 DATA R,4,G1,4,R,4,D#1,4
2090 DATA R,4,E1,4,R,4,F#1,4
2100 DATA R,4,G1,4,R,4,D#1,4
2110 DATA R,4,G3,2,E3,2,G3,4,F#3,4
2120 DATA D#3,4,E3,4,D3,2,D3,3,D3,4
2130 DATA C#3,2,C#3,2,C#3,4,C3,2,C3,2,C3
     ,4
2140 DATA B2,2,E3,2,C3,2,E3,2,B2,4,R,4
2150 DATA R,4,G2,2,E2,2,G2,4,F#2,4
2160 DATA C3,4,B2,4,G3,2,G3,2,G3,4
2170 DATA F#3,2,F#3,2,F#3,4,E3,2,E3,2,E3
     ,4
2180 DATA D#3,2,F#3,2,E3,2,F#3,2,D#3,4,R
     ,4
2190 DATA R,4,G3,2,E3,2,G3,4,F#3,4
2200 DATA D#3,4,E3,4,D3,2,D3,2,D3,4
2210 DATA C#3,2,C#3,2,C#3,4,C3,2,C3,2,C3
     ,4
2220 DATA B2,2,E3,2,C3,2,E3,2,B2,4,R,4
2230 DATA R,4,E2,2,C#2,2,E2,4,D#2,4
2240 DATA R,4,D2,2,B1,2,D2,4,C#2,4
2250 DATA R,4,C2,2,A1,2,C2,4,B1,4
2260 DATA R,4,B1,1,D#2,1,F#2,1,B2,1,E2,4
     ,B0,4
2270 :
2280 REM Voix 2
2290 DATA &38,E0,4,G0,4,E0,4,A0,4
2300 DATA E0,4,A#0,4,E0,4,A0,4
2310 DATA E0,4,G0,4,E0,4,A0,4
2320 DATA E0,4,A#0,4,E0,4,A0,4
2330 DATA E0,4,B0,4,E0,4,C1,4
2340 DATA E0,4,C#1,4,E0,4,D1,4
2350 DATA E0,4,E1,4,E0,4,F#1,4
2360 DATA E1,4,E1,4,E1,4,E0,1,D0,1,C0,1,
     B-1,1
2370 DATA A#-1,4,C1,4,A-1,4,C1,4
2380 DATA G-1,4,B0,4,F#-1,4,A#0,4
2390 DATA F#0,4,B0,4,F#0,4,C#1,4
2400 DATA B-1,4,C0,4,B-1,4,B-1,1,A-1,1,
     G-1,1,F#-1,1
2410 DATA E-1,4,B0,4,E0,4,C1,4
2420 DATA E0,4,C#1,4,E0,4,D1,4
2430 DATA E0,4,E1,4,E0,4,F#1,4
2440 DATA E1,4,E1,4,E1,4,G1,1,F#1,1,E1,1
     ,D1,1
```

```
2450 DATA C#1,4,F#0,8,F#1,1,E1,1,D#1,1,
     C#1,1
2460 DATA B0,4,E0,8,E1,1,D1,1,C#1,1,B0,1
2470 DATA A0,4,D0,8,D1,1,C1,1,B0,1,A0,1
2480 DATA G0,4,F#0,4,E0,4,B-2,4
2490 :
2500 REM Voix 3
2510 DATA &38,R,4,B0,4,R,4,C1,4
2520 DATA R,4,C#1,4,R,4,C1,4
2530 DATA R,4,B0,4,R,4,C1,4
2540 DATA R,4,C#1,4,R,4,C1,4
2550 DATA R,4,G0,4,R,4,A0,4
2560 DATA R,4,A#0,4,R,4,B0,4
2570 DATA R,4,C#1,4,R,4,D#1,4
2580 DATA G1,4,A1,4,G1,4,R,4
2590 DATA R,4,E0,4,R,4,D#0,4
2600 DATA F#2,4,E2,4,A#2,2,A#2,2,A#2,4
2610 DATA G#2,2,G#2,2,G#2,4,F#2,2,F#2,2,
     F#2,4
2620 DATA B0,4,A#0,4,F#0,4,R,4
2630 DATA R,4,G0,4,R,4,A0,4
2640 DATA R,4,A#0,4,R,4,B0,4
2650 DATA R,4,C#1,4,R,4,D#1,4
2660 DATA G1,4,A1,4,G1,4,R,4
2670 DATA R,4,A#1,2,F#1,2,A#1,4,A1,4
2680 DATA R,4,G#1,2,E1,2,G#1,4,G1,4
2690 DATA R,4,F#1,2,D1,2,F#1,4,G1,4
2700 DATA R,4,A0,4,E0,4,B-1,4
```

On a voulu que les enveloppes produisent des sons percutants, étincelants, pour rester "dans la note". D'autre part on trouvera ici, également, un très bon effet stéréo.

10

La composition sur ordinateur

Certaines compositions musicales produisent le chaos et la monotonie dont nous avons parlé au Chapitre 2. La composition tient vraiment de l'art, bien qu'il existe plusieurs méthodes et idées pour produire de la musique de manière scientifique et mathématique. Ce domaine est relativement peu exploité et c'est une occasion pour nous de découvrir de nouvelles idées - et de nouvelles compositions - avec notre ordinateur AMSTRAD.

Tout ce qui est relatif au domaine artistique, pour ne pas tomber dans l'anarchie, doit suivre quelques règles, et il en va de même pour la composition musicale. Le désordre est exprimé par le programme suivant:

```
100 REM PROGRAM 10.1
110 REM Musique pour s'amuser - Le chaos
120 :
130 WHILE -1
140 canal=INT(RND*3+1)
150 IF canal=3 THEN canal=4
160 oct=INT(RND*3+1)-1
170 note=INT(RND*12+1)
180 freq=440*(2^(oct+(note-10)/12))
190 hauteur=ROUND(125000/freq)
200 PRINT canal,hauteur
210 SOUND canal+128,hauteur,dur
220 WEND
```

Le commentaire le plus pertinent serait: "C'est intéressant!" Vous pouvez vous servir de

cette idée pour approfondir ce genre de composition et dans le domaine de la musique d'avant-garde, elle fera bonne figure. Essayez d'introduire des durées aléatoires de notes. Si vous ôtez la ligne 200, la pièce sera plus rapide.

Bien sûr, ce n'est pas ce que nous entendons par composition musicale. Notre objection est dans le fait que cette musique est complètement aléatoire, sans aucune relation entre les notes. Il faut que nous apportions un peu d'ordre dans ce chaos.

L'important était de montrer les deux extrêmes, le chaos et la monotonie, et de faire sentir la nécessité de trouver un compromis entre les deux.

L'ordinateur est absolument incapable d'émettre le moindre jugement artistique sur les notes qu'il produit et nous sommes obligés de lui spécifier ce qui produit ou non de la musique acceptable. Par contre, plus notre programme comportera de règles, plus nous nous rapprocherons d'un style et d'une certaine rigidité dans la composition. L'inspiration est produite par l'instruction RND et par de la programmation intelligente.

Le processus de la composition humaine : les algorithmes et l'heuristique

Les algorithmes sont très utilisés en informatique pour résoudre les problèmes, ils sont simplement une méthode pour arriver à une solution si elle existe. S'il n'y a pas de solution, l'algorithme le déterminera. Si un système ne résout pas un problème ou ne détermine pas qu'il n'existe aucune solution, ce n'est pas un algorithme. Les additions et soustractions mathématiques peuvent être résolues par des algorithmes parce que nous pouvons décrire exactement le cheminement des opérations.

Le procédé heuristique est souvent décrit comme un procédé empirique; on l'utilise dans les circontances où il n'y a pas d'algorithme qui puisse être valable, ou encore si l'algorithme est trop complexe. Contrairement à la méthode algorithmique, la méthode heuristique ne nous garantit pas la meilleure solution. D'ailleurs, la solution n'est même pas garantie, mais très souvent la réponse donnée arrive plus rapidement que celle de l'algorithme.

Des procédures heuristiques sont utilisées pour déterminer la stratégie des ordinateurs dans les jeux d'échecs ou de dames. Dans la stratégie de ces jeux, il est théoriquement possible d'utiliser un algorithme pour répondre à tous les coups, le nombre de solutions possibles étant un nombre fini, mais les hypothèses sont si nombreuses que nous approchons des limites d'utilisation des algorithmes.

Une approche par la méthode heuristique pourrait être de ne contrôler que le centre de l'échiquier. Nous pourrons répondre à la majorité des coups, mais nous n'avons pas l'assurance d'être gagnant. La meilleure solution, celle qui est aussi la plus utilisée, est la combinaison des deux méthodes, l'algorithmique et l'heuristique. D'ailleurs, dans le processus de réflexion humain, nous trouvons la même combinaison des deux avec en plus un grain d'inspiration...

Il existe un certain nombre de séquences, d'accords et de suites de notes qui sonnent bien et que les compositeurs connaissent. Des séquences d'accords telles que "LA mineur,SOL majeur, FA majeur, MI majeur" fournissent le fond harmonique de très nombreuses mélodies. Les compositeurs savent tout cela, mais ils sont obligés d'appliquer des règles empiriques pour greffer des mélodies par dessus... Ces règles sont le fruit de l'expérience et de l'inspiration et c'est là que nous recourons à la fonction RND et que nous inventons des

raisonnements heuristiques et des règles pour l'ordinateur.

LES DIVERS ASPECTS D'UNE COMPOSITION

Il y a trois aspects dans une composition ayant un rapport direct avec notre propos. Ce sont:

1) la mélodie
2) le rythme
3) l'harmonie

Les deux premiers sont très apparentés et la troisième, l'harmonie, est si complexe qu'il faudrait écrire plusieurs livres sur le sujet. Dans ce chapitre, nous allons examiner la mélodie et le rythme qui lui est associé, nous approfondirons l'harmonie dans le chapitre 11.

Nous passerons sur la génération aléatoire de gammes, notre étude se limitant à la composition sur ordinateur permettant de créer des mélodies plaisantes ou des séquences de notes. Nous pouvons imaginer un jeu de règles plus ou moins complexes pour réaliser cela. Ceci n'est pas trop compliqué en soi, mais les notes d'une mélodie forment une trame rythmique et nous devons également prendre en considération la longueur des notes. Nous touchons du doigt le problème le plus important: la hauteur d'une note et sa longueur sont généralement si intimement mêlées que l'un peut souvent déterminer l'autre d'un point de vue artistique.

Nous allons commencer nos expériences avec le programme suivant. Il utilise un jeu de règles se limitant à la clef de DO. Le voici:

1) la première note doit faire partie d'un accord de C majeur, soit C, E ou G.

2) L'intervalle maximal entre deux notes consécutives ne doit pas dépasser quatre notes.

3) Une note SI doit nous conduire directement à la note DO au-dessus d'elle.

4) La dernière mesure doit se terminer sur un DO et la note doit durer la longueur de la mesure.

Ces règles toutes simples vont produire un résultat bien plus agréable que le programme aléatoire précédent.

```
1000 REM PROGRAM 10.2
1010 REM Composition informatique
1020 REM basée sur des règles
1030 :
1040 GOSUB 1670:REM Mise au point
1050 :
1060 mesures=4
1070 nombre=0
1080 :
1090 FOR b=1 TO mesures
1100 PRINT "Composition d'une mesure";b
1110 GOSUB 1260:REM Composition d'une
     mesure
1120 NEXT b
1130 :
1140 FOR p=1 TO nombre
1150 PRINT air$(p),air(2,p)
1160 SOUND 1,air(1,p),air(2,p)*tempo,0,1
1170 NEXT p
1180 :
1190 PRINT "Presser barre d'espacement
     pour un autre air"
1200 PRINT "Presser R pour rejouer"
1210 com$=INKEY$:IF com$=" " THEN 1070
1220 IF UPPER$(com$)="R" THEN 1140 ELSE
     GOTO 1210
1230 END
1240 :
1250 REM Composition d'une mesure
1260 durée=0
1270 WHILE durée<>16
1280 nombre=nombre+1
1290 noteok=0
1300 WHILE noteok=0
1310 notenum=INT(RND*15+1)
```

```
1320 notes$=notes$(notenum)
1330 GOSUB 1460:REM Analyse de note
1340 GOSUB 1800:REM Règles
1350 WEND
1360 lastnote=notenum
1370 GOSUB 1530:REM Calcul de hauteur
1380 air(1,nombre)=hauteur:air$(nombre)=
     note$
1390 GOSUB 1590:REM Calcul de la durée
1400 air(2,nombre)=dur
1410 durée=durée+dur
1420 WEND
1430 RETURN
1440 :
1450 REM Analyse de note
1460 longueur=LEN(note$)
1470 IF longueur=2 THEN nom$=LEFT$(note$
     ,1):oct=VAL(RIGHT$(note$,1)):RETURN
1480 IF longueur=4 THEN nom$=LEFT$(note$
     ,2):oct=VAL(RIGHT$(note$,2)):RETURN
1490 IF MID$(note$,2,1)="-" THEN nom$=
     LEFT$(note$,1):oct=VAL(RIGHT$(note$
     ,2)) ELSE nom$=LEFT$(note$,2):oct=
     VAL(RIGHT$(note$,1))
1500 RETURN
1510 :
1520 REM Calcul de hauteur
1530 position=INSTR(gamme$,nom$)/2
1540 freq=440*(2^(oct+(position-10)/12))
1550 hauteur=ROUND(125000/freq)
1560 RETURN
1570 :
1580 REM Calcul de durée
1590 dur=2^INT(RND*2+1)
1600 :
1610 REM Dernière note semi-brève
1620 IF b=mesures THEN dur=16
1630 :
1640 IF durée+dur<=16 THEN RETURN ELSE
     1590
1650 :
1660 REM Mise au point
1670 gamme$=" C C#D D#E F F#G G#A A#B"
1680 tempo=10
1690 ENV 1,16,15,6
1700 :
```

```
1710 DIM air(2,150),air$(250)
1720 DIM notes$(15)
1730 FOR n=1 TO 15
1740 READ note$
1750 notes$(n)=note$
1760 NEXT n
1770 RETURN
1780 :
1790 DATA G-1,A-1,B-1,C0,D0,E0,F0,G0,A0,
     B0,C1,D1,E1,F1,G1
1800 :
1810 REM Règles
1820 noteok=0
1830 :
1840 REM Première note
1850 IF nombre=1 AND NOT(nom$="C" OR
     nom$="G" OR nom$="E")THEN RETURN
1860 :
1870 REM Faire B puis C
1880 IF nombre>1 AND LEFT$(air$(nombre
     -1),1)="B" THEN notenum=dernièreno
     te+1:note$=notes$(notenum):GOSUB
     1460:REM Analyse
1890 :
1900 REM Ne pas sauter plus de 4 notes
1910 IF count>1 THEN IF ABS(dernièrenote
     -notenum)>4 THEN RETURN
1920 :
1930 REM Mise au point dernière mesure
1940 IF b=mesures AND nom$<>"C" THEN
     RETURN
1950 noteok=-1
1960 RETURN
```

Commentaires

Le premier CALL que nous rencontrons nous dirige sur la routine de mise au point en 1670 pour initialiser **gamme$, tempo** et **enveloppe.** Nous avons déjà rencontré **gamme$** dans les autres programmes. En 1710 une variable **array, air,** stocke la hauteur et la durée des notes, et **air$** les noms et les octaves correspondants. Vous vous apercevrez en continuant vos expériences et en multipliant les règles qu'il est pratique d'avoir d'un côté le nom d'une note

et de l'autre sa valeur.

Note$ maintient les notes dans lesquelles le programme peut puiser. Les notes sont lues par une instruction DATA en 1790 et vous pouvez changer le choix des notes si vous le désirez.

De manière à créer un semblant d'ordre le programme compose en mesures dont le nombre est déterminé en 1060. Si vous l'incrémentez trop il faudra redimensionner **air** et **air$**.

La boucle entre 1090 et 1120 compose chaque mesure. Le travail est accompli par une routine en 1260. La variable **durée** cumule la longueur des notes pour chaque mesure et chaque mesure contient l'équivalent de seize croches. Par conséquent le programme produit une mesure 4/4 ou C. La boucle entre 1270 et 1420 tourne aussi longtemps que la durée n'est pas égale à 16. La variable **nombre** comptabilise le nombre de notes.

La petite boucle imbriquée entre 1300 et 1350 attend que la note ait passé toutes les règles en revue. La routine d'analyse en 1460 est assez familière et quand notre programme arrive en ligne 1360 elle a passé au crible toutes les règles. La hauteur est calculée par la routine en ligne 1530 et la note est stockée dans **air** et **air$**.

La routine de durée en ligne 1590 affecte une durée de 2 ou 4 à la note, et s'assure que la dernière note a une valeur de 16.

Après la composition des mesures, les lignes 1140 à 1170 se chargent de jouer la note et de l'afficher. De 1190 à 1220 vous pouvez réécouter votre composition ou en créer une autre.

La routine de hauteur et la routine d'analyse ont été séparées, de manière à pouvoir se référer à un aspect de la note sans toucher à l'autre. La routine comprenant les différentes règles a été mise en fin de programme pour

pouvoir être modifiée facilement. On présume que la note n'est pas encore ce qu'elle doit être à l'entrée de la routine en 1820. La note ne pourra sortir en 1950 que si elle a traversé toutes les règles. La première règle vérifie que la première note est bien un DO, un MI ou un SOL.

La seconde règle crée le DO qui doit suivre le SI. Ce principe peut être utilisé pour faire suivre une note par une autre, mais attention à l'arrangement des notes que vous allez obtenir. **Notenum** et **dernièrenote** se réfèrent à des positions de notes dans le flot des données. Si vous voulez que chaque SOL se transforme en LA, ajoutez 1 à G.

La règle suivante contrôle l'intervalle entre la dernière note et la note présente et rejette la note si l'intervalle est trop grand.

La dernière règle se charge de regarder si un DO termine la dernière mesure.

ESSAYONS NOTRE PROGRAMME

Bien que nous ayons fait un grand pas dans la réalisation de musique aléatoire avec ce programme, il serait tout de même honnête de dire que le résultat ne sera apprécié que par le programmeur. Un plus grand nombre de règles mélodiques est nécessaire pour réaliser quelque chose d'agréable, et le phrasé produit par la routine de durée demande un peu plus d'attention. Vous pouvez aisément adapter les règles mélodiques décrites ci-dessus. Voici quelques autres suggestions:

1) Pas plus de cinq notes ascendantes ou descendantes sans mouvement complémentaire.

2) Un SI ascendant glisse vers DO, un SI descendant vers LA.

3) Un MI ascendant glisse vers FA.

4) Un SI ne tombera jamais sur un FA et vice versa (sinon l'intervalle est trop grand, trop rude pour l'oreille.)

5) Autorisez l'introduction d'altérations mais uniquement: FA ou LA . D'autres règles sont nécessaires pour se charger de cela.

La musique consiste en une série de phrases musicales, identiques aux phrases grammaticales, qui donnent un sens, mais qui ne sont pas complètes. La routine gérant la durée n'essaye pas de réguler le contenu rythmique.

Le phrasé peut être contrôlé en passant les paramètres de durée au travers d'un jeu de règles de manière identique aux notes. De telles règles peuvent contenir les exigences suivantes:

1) Si une croche apparaît, une autre au moins doit la suivre.

2) Une mesure ne peut démarrer avec un ensemble de trois croches.

3) Il faut donner à la mesure 2 la durée de la mesure 1, à la mesure 4 celle de 3, etc. Ou encore 1 à 3 et 2 à 4.

Changez la fonction RND en ligne 1590 pour produire d'autres durées. Une autre alternative peut être l'utilisation d'un jeu prédéterminé de durée. C'est la solution la plus facile, mais il est évident que le résultat sera répétitif. Cette solution est d'une grande utilité si vous voulez vous polariser sur le seul aspect mélodique et cette solution est sûrement une amélioration de la solution aléatoire.

```
   10 REM PROGRAM 10.3
   20 REM Composition informatique
   30 REM avec rythme déterminé
   40 REM Insérer dans PROGRAM 10.2
   50 :
 1075 RESTORE 1610
```

```
1590 READ dur
1600 RETURN
1610 DATA 2,2,2,2,4,4,
1620 DATA 2,2,4,8
1630 DATA 2,2,4,2,2,4,
1640 DATA 4,4,8
1695 RESTORE 1790
1940 IF nombre=19 AND nom$<>"C" THEN
     RETURN
```

Commentaires

En 1940 on procède à la vérification du nombre de notes dans les instructions DATA. Vous pouvez ajouter d'autres variantes en permettant l'échange avec des durées prédéterminées. Vous pourrez ainsi profiter de tous les avantages.

Si vous ajoutez davantage de règles, vous affecterez le style de la composition et si le nombre de règles est suffisant vous allez créer votre propre style, un style unique. Il y a d'autres moyens de programmation du style dans une composition et nous allons examiner une de ces méthodes.

L'analyse de la note en composition

Si nous procédons à l'analyse d'une composition musicale, en dressant un tableau précisant le nombre d'apparitions de chaque note, nous aurons une analyse du premier ordre. Si par la suite nous écrivons un programme qui joue les notes de ce même tableau en fonction de leur fréquence d'apparition nous aurons sûrement une composition qui ira dans le sens de la musique que nous venons d'analyser.

L'idée n'est pas nouvelle et des compositeurs comme Stephen Foster (qui a écrit "Oh Susannah", entre autres) s'y consacrent depuis une vingtaine d'années.

Le succès de la méthode dépend de la composition que l'on veut analyser. Si les notes apparaissent dans des proportions égales le résultat n'est pas bien fameux. En fait, si nous nous basons sur ce que nous appelons une analyse du premier ordre, nous obtiendrons toujours des mélodies ressemblant à de la musique aléatoire. Il faut absolument prendre en considération la durée de la note et effectuer une analyse similaire.

Nous pouvons resserrer notre analyse en comptabilisant le nombre de fois où telle note est suivie par telle autre. Nous appellerons cela une analyse du deuxième ordre. Il est possible de prendre en considération d'autres paramètres et de procéder à des analyses du troisième ou quatrième ordre. Plus nous incrémentons l'ordre d'analyse, meilleur sera le résultat, mais il y a des chances qu'en poussant trop loin, on produise un morceau qui ressemble de très près à la composition originale.

Le prochain programme procèdera à des analyses du premier, du second et du troisième ordre sur une mélodie se trouvant dans les instructions DATA. Par la suite il composera une mélodie basée sur son analyse, il va de soi que cette mélodie pourra changer à chaque nouveau passage dans le programme.

```
1000 REM PROGRAM 10.4
1010 REM Compositions informatiques
1020 REM basées sur analyse des notes
1030 :
1040 MODE 1
1050 DEFINT a-z
1060 DIM air$(238),dur(238),f1(36)
1070 gamme$="  C-1 C#-1D-1 D#-1E-1 F-1
     F#-1G-1 G#-1A-1 A#-1B-1 C0 C#0 D0
     D#0 E0 F0 F#0 G0 G#0 A0 A#0 B0 C1
     C#1 D1 D#1 E1 F1 F#1 G1 G#1 A1 A#1
     B1 "
1080 WINDOW #0,1,40,2,40
1090 WINDOW #1,20,35,1,1
1100 ENV 1,1,15,1,1,0,6,1,-2,2,13,-1,8
```

```
1110 ENT -1,1,1,3,2,-1,3,1,1,3
1120 tempo=6
1130 :
1140 GOSUB 1690:REM Donne le ton
1150 GOSUB 1840:REM Nouvelle gamme
1160 GOSUB 2080:REM Analyse du morceau
1170 GOSUB 2220:REM Calcul des
     pourcentages
1180 GOSUB 2530:REM Sortie imprimée
1190 :
1200 penult=INT(RND*longamme+1)
1210 dernièrenote=INT(RND*longamme+1)
1220 PRINT "Entrer la profondeur de
     recherche (1/2/3)  Ceci peut","être
     changé pendant que le programme
     tourne"
1230 jouer=VAL(INKEY$):IF jouer<1 OR
     jouer>3 THEN 1230
1240 CLS:PRINT#1,"Profondeur :";jouer
1250 :
1260 d=1
1270 WHILE -1
1280 GOSUB 1360:REM Choisir une note
1290 GOSUB 1600:REM Jouer une note
1300 p1=VAL(INKEY$):IF p1>0 AND p1<4
     THEN jouer=p1:PRINT#1,"Profondeur
     :";jouer
1310 WEND
1320 :
1330 END
1340 :
1350 REM Choisir une note
1360 dice=INT(RND*100+1)
1370 note=0:sum=0
1380 WHILE sum<dice
1390 note=note+1
1400 REM Au cas où une note n'a jamais
1410 REM suivi une séquence
1420 REM particulière de notes
1430 IF note>longamme THEN note=INT(RND*
     longamme+1):sum=dice:GOTO 1450
1440 ON jouer GOSUB 1530,1550,1570
1450 WEND
1460 :
1470 note$=MID$(gamme2$,note*4,4)
1480 penult=dernièrenote:dernièrenote=
     note
```

```
1490 REM PRINT note$,dur(d)
1500 RETURN
1510 :
1520 REM Arrangement n°1
1530 sum=sum+f1(note):RETURN
1540 REM Arrangement n°2
1550 sum=sum+f2(dernierenote,note):
     RETURN
1560 REM Arrangement n°3
1570 sum=sum+f3(penult,dernierenote,note
     ):RETURN
1580 :
1590 REM Jouer une note
1600 position=INSTR(gamme$,note$)/4-12
1610 freq=440*(2^(oct+(position-10)/12))
1620 hauteur=ROUND(125000/freq)
1630 SOUND 1,hauteur,dur(d)*tempo,0,1,1
1640 d=d+1
1650 IF d>longuair THEN d=1
1660 RETURN
1670 :
1680 REM Prendre le morceau
1690 CLS:PRINT "Lecture du morceau pour
 analyse ..."
1700 REM RESTORE au morceau défini
1710 RESTORE 2780:REM 2780,2850,2920
1720 :
1730 nombre=0
1740 WHILE note$<>"X"
1750 READ note$,dur:IF note$="X" THEN
     1790
1760 nombre=nombre+1
1770 air$(nombre)=note$
1780 dur(nombre)=dur
1790 WEND
1800 longuair=nombre
1810 RETURN
1820 :
1830 REM Nouvelle gamme
1840 PRINT "Calcul nouvelle gamme..."
1850 :
1860 FOR n=1 TO longuair
1870 npos=INSTR(gamme$,air$(n))/4
1880 f1(npos)=f1(npos)+1
1890 NEXT n
1900 :
```

```
1910 gamme2$="    "
1920 :
1930 FOR n=1 TO 36
1940 IF f1(n)>0 THEN gamme2$=gamme2$+
     MID$(gamme$,n*4,4)
1950 NEXT n
1960 :
1970 PRINT "Nouvelle gamme:":PRINT:PRINT
     gamme2$:PRINT
1980 longamme=(LEN(gamme2$)-3)/4
1990 PRINT "Longueur de la gamme =";
     longamme:PRINT
2000 :
2010 DIM f2(longamme,longamme),f3
     (longamme,longamme,longamme)
2020 :
2030 REM Correction de l'arrangement f1
2040 FOR n=1 TO 36:f1(n)=0:NEXT n
2050 RETURN
2060 :
2070 REM Analyse de l'air
2080 PRINT "Analyse de l'air..."
2090 FOR n=1 TO longuair
2100 pos1=INSTR(gamme2$,air$(n))/4
2110 f1(pos1)=f1(pos1)+1
2120 IF n>longuair-1 THEN 2150
2130 pos2=INSTR(gamme2$,air$(n+1))/4
2140 f2(pos1,pos2)=f2(pos1,pos2)+1
2150 IF n>longuair-2 THEN 2180
2160 pos3=INSTR(gamme2$,air$(n+2))/4
2170 f3(pos1,pos2,pos3)=f3(pos1,pos2,
     pos3)+1
2180 NEXT n
2190 RETURN
2200 :
2210 REM Calcul des pourcentages
2220 PRINT "Calcul de la fréquence du
     premier arrangement..."
2230 sum1=0
2240 FOR n1=1 TO longamme
2250 sum1=sum1+f1(n1)
2260 NEXT n1
2270 FOR n1=1 TO longamme
2280 f1(n1)=f1(n1)*100/sum1
2290 NEXT n1
2300 :
```

```
2310 PRINT "Calcul de la fréquence du
     deuxième arrangement..."
2320 FOR n1=1 TO longamme
2330 sum2=0
2340 FOR n2=1 TO longamme
2350 sum2=sum2+f2(n1,n2)
2360 NEXT n2
2370 IF sum2>0 THEN FOR n2=1 TO longamme
     :f2(n1,n2)=f2(n1,n2)*100/sum2:NEXT
     n2
2380 NEXT n1
2390 :
2400 PRINT "Calcul de la fréquence du
     troisième arrangement..."
2410 FOR n1=1 TO longamme
2420 FOR n2=1 TO longamme
2430 sum3=0
2440 FOR n3=1 TO longamme
2450 sum3=sum3+f3(n1,n2,n3)
2460 NEXT n3
2470 IF sum3>0 THEN FOR n3=1 TO longamme
     :f3(n1,n2,n3)=f3(n1,n2,n3)*100/sum3
     :NEXT n3
2480 NEXT n2
2490 NEXT n1
2500 RETURN
2510 :
2520 REM Impression
2530 PRINT:PRINT "Voulez-vous une sortie
     imprimée(O/N)?"
2540 com$=INKEY$:IF UPPER$(com$)="N"
     THEN PRINT :RETURN ELSE IF UPPER$
     (com$)<>"O" THEN 2540
2550 :
2560 FOR n1=1 TO longamme
2570 IF f1(n1)>0 THEN PRINT MID$(gamme2$
     ,n1*4,4);"...";f1(n1)
2580 NEXT n1
2590 :
2600 FOR n1=1 TO longamme
2610 FOR n2=1 TO longamme
2620 IF f2(n1,n2)>0 THEN PRINT MID$
     (gamme2$,n1*4,4);"- ";MID$(gamme2$,
     n2*4,4);"...";f2(n1,n2)
2630 NEXT n2
2640 NEXT n1
```

```
2650 :
2660 FOR n1=1 TO longamme
2670 FOR n2=1 TO longamme
2680 FOR n3=1 TO longamme
2690 IF f3(n1,n2,n3)>0 THEN PRINT MID$
     (gamme2$,n1*4,4);"- ";MID$(gamme2$,
     n2*4,4);"- ";MID$(gamme2$,n3*4,4);
     "...";f3(n1,n2,n3)
2700 NEXT n3
2710 NEXT n2
2720 NEXT n1
2730 PRINT
2740 :
2750 RETURN
2760 :
2770 REM Hymne à la joie - Beethoven
2780 DATA E0,8,E0,8,F0,8,G0,8,G0,8,F0,8,
     E0,8,D0,8,C0,8,C0,8,D0,8,E0,8,E0,12
2790 DATA D0,4,D0,16,E0,8,E0,8,F0,8,G0,8
     ,G0,8,F0,8,E0,8,D0,8,C0,8,C0,8,D0,8
2800 DATA E0,8,D0,12,C0,4,C0,16,D0,8,D0,
     8,E0,8,C0,8,D0,8,E0,4,F0,4,E0,8,C0,
     8
2810 DATA D0,8,E0,4,F0,4,E0,8,D0,8,C0,8,
     D0,8,G-1,16,E0,8,E0,8,F0,8,G0,8,G0,
     8
2820 DATA F0,8,E0,8,D0,8,C0,8,C0,8,D0,8,
     E0,8,D0,12,C0,4,C0,16,X,0
2830 :
2840 REM Jésus, que ma joie demeure
     - Bach
2850 DATA G0,6,A0,6,B0,6,D1,6,C1,6,C1,6,
     E1,6,D1,6,D1,6,G1,6,F#1,6,G1,6,D1,6
2860 DATA B0,6,G0,6,A0,6,B0,6,C1,6,D1,6,
     E1,6,D1,6,C1,6,B0,6,A0,6,B0,6,A0,6,
     B0,6,G0,6
2870 DATA F#0,6,G0,6,A0,6,D0,6,F#0,6,A0,
     6,C1,6,B0,6,A0,6,B0,6,G0,6,A0,6,B0,
     6
2880 DATA D1,6,C1,6,C1,6,E1,6,D1,6,D1,6,
     G1,6,F#1,6,G1,6,D1,6,B0,6,G0,6,A0,6
2890 DATA B0,6,E0,6,D1,6,C1,6,B0,6,A0,6,
     G0,6,D0,6,G0,6,F#0,6,G0,18,X,0
2900 :
2910 REM La Lavandière Irlandaise, Gigue
```

```
2920 DATA D0,4,B-1,4,G-1,4,G-1,4,D-1,4,
     G-1,4,G-1,4,B-1,4,G-1,4,B-1,4,D0,4,
     C0,4,B-1,4
2930 DATA C0,4,A-1,4,A-1,4,E-1,4,A-1,4,
     A-1,4,C0,4,A-1,4,C0,4,E0,4,D0,4,C0,
     4
2940 DATA B-1,4,G-1,4,G-1,4,D-1,4,G-1,4,
     G-1,4,B-1,4,G-1,4,B-1,4,D0,4,C0,4,
     B-1,4
2950 DATA C0,4,B-1,4,C0,4,A-1,4,D0,4,C0,
     4,B-1,4,G-1,4,G-1,4,G-1,8,G0,4
2960 DATA G0,4,D0,4,G0,4,G0,4,D0,4,G0,4,
     G0,4,D0,4,G0,4,B0,4,A0,4,G0,4
2970 DATA F#0,4,D0,4,F#0,4,F#0,4,D0,4,
     F#0,4,F#0,4,D0,4,F#0,4,A0,4,G0,4,
     F#0,4
2980 DATA E0,4,G0,4,G0,4,D0,4,G0,4,G0,4,
     C0,4,G0,4,G0,4,B-1,4,G0,4,G0,4
2990 DATA C0,4,B-1,4,C0,4,A-1,4,D0,4,C0,
     4,B-1,4,G-1,4,G-1,4,G-1,8,X,0
3000 d=1:WHILE -1:note$=air$(d):GOSUB
     1600:WEND
```

Commentaires

L'analyse de notes est une opération qui sied parfaitement bien à un ordinateur. La première et la seconde analyse sont très rapides et n'utilisent que peu de mémoire, mais dès que nous passons à l'analyse du troisième ordre, il faudra garder en mémoire la trace de trois éléments consécutifs. Il faudra dimensionner vos **array** de la manière suivante:

```
DIM   f3(36,36,36)
```

pour une gamme de trois octaves où même plus large, et là nous aurons des problèmes de taille mémoire. En regardant de plus près nous pouvons voir que toutes les notes ne sont pas utilisées et qu'il y a un gâchis d'emplacements en mémoire. Un moyen pour contourner ce problème consiste en un calcul d'une nouvelle gamme basée sur les notes utilisées dans la composition de départ pour dimensionner les **array** une fois que

l'on a recensé tous les besoins.

La variable **air$** contient les notes et **dur** est dimensionnée pour contenir les durées. **f1** puis par la suite **f2** et **f3** gardent trace des séquences dans les trois types d'analyse.

La gamme des notes disponibles est mise dans **gamme$** en 1070. Si vous entrez d'autres notes, vérifiez qu'elles ne soient pas en dehors de cette gamme ou alors adaptez **gamme$**.

La 1140 fait appel à la routine en ligne 1690 qui procède à la récupération des données dans l'instruction DATA. Si vous êtes obligé de mettre en place plusieurs lignes de DATA, veillez à mettre en place, et correctement, une instruction RESTORE. Les données se terminent par un X et la variable **longuair** contient le nombre de notes à lire.

La prochaine routine qui sera appelée est la routine qui calcule la nouvelle gamme. Les notes sont lues, comparées avec celles de **gamme$** et mises dans **gamme2$** qui contient uniquement les notes utilisées. Les **array** pour stocker les résultats des analyses de second et troisième ordre sont dimensionnées en 2010. L'array **f1** est remis à zéro pour une utilisation future.

La routine en 2080 passe de note en note et comptabilise la répétition des notes. **f1** compte les notes, **f2** les doublets, **f3** les triplets.

La routine située en 2220 calcule chaque ordre d'analyse, l'un après l'autre. Elle détermine premièrement le nombre de fois qu'une séquence apparaît dans un morceau, puis les valeurs de **f1**, **f2** et **f3** sont ajoutées pour voir leur nombre total. On repasse une nouvelle fois dans la boucle pour assigner une nouvelle valeur en pourcentage. Examinez le premier exemple entre les lignes 2230 et 2290. Le second et le troisième ordre utilisent une série de boucles imbriquées pour vérifier toutes les combinaisons, mais seul un pourcentage est

calculé s'il le faut, aux lignes 2370 à 2470. Cette méthode réutilise les **array** pour gagner de la place-mémoire.

Nous pouvons avoir quelques doutes sur les valeurs de pourcentages puisque nous utilisons des variables entières et que certaines lignes comme la 2280 produisent quelquefois des valeurs décimales. En, fait le résultat est assez fiable et on a programmé de sorte à utiliser un minimum de mémoire et à obtenir un temps d'exécution correct. Vous pouvez modifier ces lignes pour avoir des valeurs exactes de pourcentage et au moins des arrondis corrects.

Vous pouvez utiliser le même principe pour calculer des analyses de niveau supérieur. Le résultat sera intéressant et vous pourrez vous rendre compte dès le troisième degré de la similitude avec la mélodie originale.

La routine d'affichage en 2530 utilise des boucles imbriquées, de la même manière que la routine de calcul du pourcentage.

Si vous découvrez trois notes suivies de la valeur 100, vous saurez qu'après les deux premières notes la troisième apparaît à chaque fois. Si à la place de 100 nous n'avions que 50, ceci nous indiquerait que la répétition de la troisième se fait à 50%.

Avant de commencer à composer le programme a besoin de deux notes de base sur lesquelles il base ses calculs. Ces notes sont déterminées de manière aléatoire aux lignes 1200 et 1210. En 1220 on vous demande quel type d'analyse vous voulez.

La boucle entre 1270 et 1310 joue la mélodie. En 1360 nous trouvons la routine chargée de décider de la note à jouer en accord avec les pourcentages d'apparition. La variable **dice** est un pourcentage aléatoire et **sum** est incrémentée en 1530, 1550 ou 1570 suivant la profondeur d'analyse.

Vous pouvez désirer un peu plus d'explications car **note, dernièrenote** et **penult** ont des valeurs servant à fournir des notes ou des séquences de notes dans les trois array : **f1, f2** et **f3.** Vous pouvez voir ci-dessous un exemple du contenu de l'array f2.

```
f2(dernièrenote,1)=20

f2(dernièrenote,2)=50

f2(dernièrenote,3)=10

f2(dernièrenote,4)=8

f2(dernièrenote,5)=12
```

Si **dice** est égal à 72 la routine se déroulera de la manière suivante: **sum** va démarrer avec une valeur 0 et **note** avec une valeur 1. Nous ajoutons **f2 (dernièrenote,note)** à **sum.** Si la valeur est égale ou supérieure à dice nous quittons la boucle. Dans ce cas, si note égale 1 sum sera égale à 20 ce qui n'est pas suffisant, donc note sera incrémentée de 1 et nous recommençons. Cette fois 50 est ajouté à sum qui n'est encore pas suffisant et nous répétons le processus jusqu'à ce que la note soit égale à 3, ainsi la valeur sera égale à 80, donc plus grande que dice et nous sortirons de la boucle.

La variable **note$** est calculée à partir de **note. dernièrenote** et **penult** sont ajustées et nous nous dirigeons vers la routine d'exécution en 1600.

Un REM de 1490 a été mis en place pour ne pas ralentir le programme avec un PRINT, mais pour la mise au point vous pouvez l'ôter. La routine en 1360 peut également demander un certain temps d'exécution.

La routine d'émission de la note est toujours la même. Remarquez que la variable position nous retourne des valeurs de 1 à 36,

qui est le nombre de notes dans **gamme$**. La variable **oct** n'est définie nulle part et elle est égale à 0. Nous calculerons la fréquence sans tenir compte de la valeur **oct**. Toutes les variables numériques étant définies comme variables entières, le résultat du calcul de la fréquence et de la hauteur sera une valeur entière approchant de la valeur réelle, mais la différence sera minime.

En 1260, la variable **d** est utilisée pour accéder aux durées contenues dans l'array **dur**, et si sa valeur dépasse la longueur, elle est remise à 1 en 1650.

Le programme ne reproduira pas exactement la mélodie originale (c'est théoriquement possible). Pour y arriver appuyez sur la touche Break dès que le programme a passé le stade de la lecture des notes et entrez:

```
GOTO 3000
```

Vous pouvez aussi utiliser cette méthode pour voir si les notes ont été correctement mises en place.

Essayez votre programme

Les morceaux qui fournissent les instructions DATA ont été choisis en raison de la durée des notes qui est à peu près uniforme. Ceci nous permet de comparer plus facilement notre composition avec l'air original.

Il est intéressant de jouer ces compositions à d'autres personnes et de leur demander de deviner l'origine de la mélodie. Si vous utilisez plusieurs airs du même compositeur vous devriez avoir une composition dans le style, mais complètement nouvelle. Vous pouvez également essayer plusieurs airs de rock'n'roll; généralement, ils utilisent tous les mêmes accords de base et la même structure harmonique.

Le programme produit les effets les plus intéressants lorsqu'il est utilisé de cette manière. En approvisionnant votre programme avec un grand flot de données, vous comprendrez l'importance du gain de place-mémoire. D'ailleurs, si les données sont très longues il vous faudra peut-être augmenter la taille des éléments en 1060. Vous pouvez également utiliser les données des programmes du Chapitre 9 (le Rondo turc, etc.) pour passer à l'analyse. Attention,il vous faudra dans certains cas effectuer des changements dans **gamme$.**

Un programme pour analyser tout le morceau

A partir des principes que nous venons de discuter et de démontrer, il est possible d'envisager un programme pouvant analyser plusieurs parties d'un air de musique. Il y a au moins deux manières d'aborder le problème. L'une a été suggérée et demande à ce que les durées passent par des routines similaires aux routines des notes. Nous aurons une variété de rythmes basés sur les rythmes de l'air d'origine, mais sans rapport avec les notes.

La seconde méthode propose une analyse des notes et de leurs durées respectives, de manière à ce qu'un DO1 de durée 2 soit traité de manière différente d'un DO1 de durée 4. Cette solution est presque parfaite, le seul inconvénient est qu'elle est très gourmande en mémoire et pratiquement impossible à réaliser.

Toujours dans la même ligne de pensée, il est possible également d'inclure une analyse des harmonies indiquées par la structure des accords. La plupart des chansons modernes changent d'accord à toutes les mesures, ou toutes les quatre mesures, et nous pouvons analyser cela facilement.

Si vous décidez d'analyser des airs de rock'n'roll, il n'est pas nécessaire d'analyser les accords puisqu'ils sont pratiquement tous

identiques. Dans le cas d'une clé de DO, les mesures contiendront les accords suivants:

DO/DO/DO/DO/FA/FA/DO/DO/SOL/FA/DO/DO

Ceci est le fameux blues à douze mesures. Tous les accords sont des accords majeurs, bien qu'ils soient souvent joués en septième, et beaucoup de notes seront jouées autour des septièmes (dans le cas de DO7, SI bémol). S'il y a un solo instrumental, la dernière mesure de DO est souvent remplacée par une mesure de SOL (ou SOL7).

Dans un programme idéal, en même temps que nous analysons notes et durées nous devrions également examiner les accords et analyser ces trois combinaisons comme un seul élément. L'intérêt d'un tel programme n'est plus à démontrer et la programmation ne devrait pas être trop difficile. Par contre, pour faire tenir le programme dans la place qui nous est allouée il faudra faire preuve d'ingéniosité, et presque de virtuosité.

11

Votre composition sur Amstrad

Dans le dernier chapitre, nous allons examiner quelques-unes des difficultés qu'il nous faudra surmonter pour produire des mélodies. Tous les programmes que nous avons vus produisent des mélodies, mais un de leurs défauts principaux est le manque d'harmonie.

L'harmonie est généralement le fruit de plusieurs notes exécutées au même instant. En fait, la génération d'harmonies ne pose aucun problème, mais la production d'harmonies plaisantes pour notre ouïe est autrement difficile.

La structure harmonique (ou le fond harmonique) d'une pièce de musique peut être produite par un accompagnement de guitare ou par des accords de piano ou d'orgue. Le programme 2.1 du Chapitre 2 nous donne des exemples d'harmonies produites par différents accords.

Pour une progression d'accords déterminée il est possible de produire un nombre de mélodies indéterminé. Réciproquement, on peut choisir un nombre illimité d'accords en accompagnement d'une mélodie donnée. En fait chaque note de la mélodie pourrait s'accommoder d'un accord différent des autres notes, et encore plusieurs accords pourraient convenir pour chaque note. Les accords peuvent changer à chaque mesure, deux fois par mesure, ou toutes les huit mesures. Quelques-unes de ces combinaisons pourront nous ennuyer et nous paraître

franchement désagréables. Tout est possible, il suffit de trouver une personne pour l'apprécier, même si ce n'est que le compositeur... La majeure partie de la musique suit tout de même des règles de structure harmonique qui la rendent plus agréable.

La structure harmonique des chansons populaires

La plupart des airs populaires dépendent fortement de la progression des accords pour plaire; certains musiciens sont capables de prendre une mélodie et d'adapter un accompagnement à cet air. Il peut différer de l'original; il en sera pourtant assez proche pour que la plupart des gens reconnaissent la mélodie. Il est peu probable que deux musiciens harmonisent un air exactement de la même manière, et aucun critère ne détermine quelle adaptation est la meilleure par rapport à l'original.

Le type d'accords utilisés dans un morceau est indicatif du niveau d'harmonies que nous, public, avons atteint. Les ménestrels d'antan utilisaient des jeux d'accords beaucoup plus simples, en accord avec le niveau d'appréciation (ou de tolérance) des harmonies atteint par le public en ce temps-là. Aujourd'hui nous sommes plus tolérants, et des accords plus complexes, des harmonies plus dissonantes sont acceptés (consonant est l'inverse de dissonant. Un accord dissonant peut être produit par un son de deux notes distantes d'un demi-ton).

Les musiciens de jazz sont connus pour leurs improvisations dans les mélodies et les accords. Très souvent ils dérivent délibérément du son original, ce qui peut amener une certaine difficulté à comprendre le jazz.

La musique classique a ses propres règles et ses propres accords (la plupart des grands musiciens ont passé outre) et vous pouvez vous rendre compte de la manière dont nous avons progressé, au point de vue de l'harmonie, si

vous écoutez des pièces de musique des compositeurs britanniques PURCELL ou ARNE (auteur de "Rule Britannia").

Dans un genre de musique bien défini votre oreille s'attend à un certain type d'accords et d'arrangements de mélodies. Un programme pour produire des airs populaires sera complexe. Il sera plus facile de produire de la musique dans un style classique qui, apparemment, demande une structure et une harmonie plus simples.

Produire des résultats acceptables

Dans le domaine de la composition par ordinateur, il faut être modeste et si nous arrrivons à quelque chose qui donne des sons plaisants et qui ne produit pas de haut-le-corps chez l'auditeur nous pouvons parler de succès. Ce chapitre va un peu plus loin en nous permettant de programmer l'AMSTRAD sur trois voix au maximum, ce qui est musicalement acceptable.

En préjugeant des problèmes que nous allons rencontrer, vous allez penser que générer des morceaux de musique à deux ou trois voix est deux ou trois fois plus difficile. Vous auriez sûrement raison si nous voulions suivre toutes les règles académiques, tous les standards. De même, si nous tentions de composer des pièces avec un canon classique à trois voix, nous aurions des problèmes, quoique des règles, ou du moins une certaine logique, pourraient être trouvées et formulées de manière un peu plus rigide.

Notre but principal sera de produire une série de deux ou trois notes émettant une tonalité consonante et qui forment ensemble un accord plaisant et mélodique.

Composition d'accords aléatoires

Après toutes ces discussions sur les règles et les compositions, il est vraiment difficile d'expliquer exactement ce qui rend un morceau de musique bon ou mauvais, plaisant ou déplaisant. Chaque composition nécessite un élément aléatoire, sinon nous jouerions une musique pré-programmée qui n'aurait aucune originalité. Notre problème est de trouver une manière de contrôler les paramètres appliqués au choix aléatoire des notes.

La méthode la plus rapide et la plus facile est d'utiliser la fonction "random" pour faire un choix dans une série de notes, connues pour s'accorder facilement entre elles. Les résultats sont excellents mais on se doute qu'ils manqueront un peu de piquant.

ETRE MOZART TOUT DE SUITE

Mozart est réputé pour avoir imaginé un système de composition basé sur le jeu de dés. Sa méthode peut facilement être transcrite sur ordinateur et produira sûrement des résultats spectaculaires; il faudra cependant réaliser une grande partie de la composition soi-même.

Les dés étaient utilisés pour choisir un nombre de mesures qui avaient été composées précédemment. Par exemple, pour composer un air comportant huit mesures nous dessinerons un tableau de huit colonnes sur six rangées, chacune contenant une mesure de musique. Le premier jet de dé sélectionnera une mesure de la première colonne, le second jet une mesure de la deuxième colonne, etc.

Le gros problème reste la composition des 8x6, soit 48 mesures qui doivent pouvoir se suivre les unes les autres. On suppose que Mozart devait avoir plus de facilités que nous pour ce genre de composition.... Si le coeur vous en dit et que la chose ne vous semble pas

si difficile que cela, le résultat sera certainement très bon, mais dans ce cas c'est à vous que revient la composition et non à l'ordinateur.

Il est possible d'appliquer d'autres règles à un programme de composition, par exemple insérer périodiquement une série de notes prédéfinies dans un programme aléatoire, mais les résultats ne sont guère probants.

Il faudrait pouvoir intervenir sur le choix aléatoire de l'ordinateur, mais sans pour cela imposer notre propre composition. Comme un ordinateur ne connaît rien des combinaisons de notes qui produisent de bons résultats, nous pouvons peut-être l'aider en lui indiquant une liste des notes qui lui sont permises.

Utilisation des accords en tant que base de composition

Une manière d'élargir notre sélection est d'établir une relation entre le choix des notes et des accords, en tenant compte des notes qui constituent ces accords. L'ordinateur choisira ses notes dans un registre prédéfini. Le programme 11.1 réalise cela et nous permet de programmer l'AMSTRAD avec n'importe quelle séquence d'accords et avec le type d'accord de votre choix.

```
1000 REM PROGRAM 11.1
1010 REM Compositions Informatiques
1020 REM Basées sur une suite d'accords
1030 :
1040 MODE 1
1050 DEFINT a-e,g-z
1060 gamme$=" C C#D D#E F F#G G#A A#B"
1070 registredaccords$="   M   7   9
     min min6min7min9maj6maj7aug dim"
1080 :
1090 DIM note$(11,6)
1100 :
1110 REM RESTORE aux DATA accords voulus
```

```
1120 RESTORE 2390:REM 2390,2420,2450,
     2520
1130 READ nombraccords
1140 DIM mélodie$(nombraccords)
1150 :
1160 FOR n=1 TO nombraccords
1170 READ mélodie$(n)
1180 NEXT n
1190 :
1200 RESTORE 2160
1210 FOR n=1 TO 11
1220 FOR c=1 TO 6
1230 READ note$(n,c)
1240 NEXT c
1250 NEXT n
1260 :
1270 ENV 1,1,14,10,4,-1,6,1,0,10,10,-1,
     12
1280 ENT 1
1290 ENV 2,1,15,20,1,0,20,4,-1,10,11,-1,
     12
1300 ENT -2,1,2,3,2,-2,3,1,2,3
1310 ev1=1:ev2=1:ev3=2
1320 :
1330 PRINT:PRINT:PRINT
1340 INPUT "Entrer le nombre de temps
     dans la mesure";nombraccords
1350 INPUT "Entrer le tempo (25 ou plus)
     ";tempo1
1360 tempo=tempo1
1370 :
1380 PRINT "Voulez-vous un changement de
     rythme(O/N)?";
1390 sync$=UPPER$(INKEY$):IF sync$="O"
     THEN sync=-1 ELSE IF sync$="N" THEN
     sync=0 ELSE GOTO 1390
1400 PRINT sync$
1410 :
1420 opus=0
1430 WHILE -1
1440 opus=opus+1
1450 in0=INT(RND*26):in1=INT(RND*26):IF
     in0=in1 THEN 1450
1460 INK 0,in0:INK 1,in1:BORDER INT(RND*
     26)
1470 LOCATE 12,10:PRINT "Composition
     d'un opus";opus
```

```
1480 :
1490 FOR n=1 TO nombraccords
1500 syncpoint=INT(RND*4+1)
1510 FOR temps=1 TO nombraccords*2
1520 IF sync THEN GOSUB 1970:REM Rhythme
1530 GOSUB 1610:REM Jouer
1540 NEXT temps
1550 NEXT n
1560 WEND
1570 :
1580 END
1590 :
1600 REM Jouer
1610 GOSUB 1740:REM Analyse des accords
1620 n$=note1$:GOSUB 1880:REM Faire une
     note
1630 SOUND &38+1,hauteur,tempo,0,ev1,ev1
1640 n$=note2$:GOSUB 1880:REM Faire une
     note
1650 SOUND &38+4,hauteur,tempo,0,ev2,ev2
1660 REM Basse
1670 position=INSTR(gamme$,ke$)/2
1680 freq=440*(2^(-1+(position-10)/12))
1690 hauteur=ROUND(125000/freq)
1700 SOUND &38+2,hauteur,tempo,0,ev3,ev3
1710 RETURN
1720 :
1730 REM Analyse des accords
1740 accord$=mélodie$(n)
1750 IF MID$(accord$,2,1)="#" THEN ke$=
     LEFT$(accord$,2):genredaccord$=MID$
     (accord$,3) ELSE ke$=LEFT$(accord$,
     1):genredaccord$=MID$(accord$,2)
1760 ke=INSTR(gamme$,ke$)/2
1770 numaccord=INSTR(registredaccords$,
     genredaccord$)/4
1780 :
1790 choix1=INT(RND*6+1)
1800 choix2=INT(RND*6+1)
1810 IF choix2=choix1 THEN 1800
1820 :
1830 note1$=notes$(numaccord,choix1)
1840 note2$=notes$(numaccord,choix2)
1850 RETURN
1860 :
1870 REM Choisir une note
```

```
1880 IF LEN(n$)=2 THEN nom$=LEFT$(n$,1)
     ELSE nom$=LEFT$(n$,2)
1890 oct=VAL(RIGHT$(n$,1))
1900 position=INSTR(gamme$,nom$)/2+ke-1
1910 freq=440*(2^(-1+(position-10)/12))
1920 hauteur=ROUND(125000/freq)
1930 RETURN
1940 :
1950 REM Variations de rythme
1960 REM GOTO Rhythme voulu
1970 ON syncpoint GOTO 2000,1990,2020,
     2050
1980 :
1990 IF temps=1 THEN tempo=tempo1*1.5
     ELSE IF temps=2 THEN tempo=tempo1*
     0.5 ELSE tempo=tempo1
2000 RETURN
2010 :
2020 IF temps=nombraccords OR temps=
     nombraccords+1 THEN tempo=tempo1*
     0.5 ELSE IF temps=1 THEN tempo=
     tempo1*2 ELSE tempo=tempo1
2030 RETURN
2040 :
2050 IF temps=nombraccords OR temps=
     nombraccords+1 OR temps=nombraccords
     +2 OR temps=nombraccords+3 THEN
     tempo=tempo1*0.5:GOSUB 1610 ELSE
     tempo=tempo1
2060 RETURN
2070 :
2080 IF temps MOD 2=1 THEN tempo=tempo1*
     0.75 ELSE IF temps MOD 2=0 THEN
     tempo=tempo1*0.25
2090 RETURN
2100 :
2110 REM Non Sync - temps 9/8
2120 IF temps=1 THEN tempo=tempo1*2 ELSE
     IF temps=2 THEN temps=3 ELSE tempo=
     tempo1
2130 RETURN
2140 :
2150 REM Majeur
2160 DATA G0,C1,E1,G1,C2,E2
2170 REM Septième
2180 DATA A#0,C1,E1,G1,A#1,C2
```

```
2190 REM Neuvième Majeur
2200 DATA D1,E1,G1,A#1,C2,D2
2210 REM Mineur
2220 DATA G0,C1,D#1,G1,C2,D#2
2230 REM Sixième Mineur
2240 DATA A0,C1,D#1,G1,A1,C2
2250 REM Septième Mineur
2260 DATA A#0,C1,D#1,G1,A#1,C2
2270 REM Neuvième Mineur
2280 DATA D1,D#1,G1,A#1,C2,D2
2290 REM Sixième Majeur
2300 DATA A0,C1,E1,G1,A1,C2
2310 REM Septième Majeur
2320 DATA B0,C1,E1,G1,B1,C2
2330 REM Augmenté
2340 DATA G#0,C1,E1,G#1,C2,E2
2350 REM Diminué
2360 DATA D#1,F#1,A1,C2,D#2,F#2
2370 :
2380 REM DATA air commence ici
2390 DATA 12
2400 DATA C7,C7,C7,C7,F7,F7,C7,C7,G7,F7,
     C7,G7
2410 :
2420 DATA 12
2430 DATA F7,F9,F7,F9,A#7,A#9,F7,F9,C9,
     A#9,F9,C7
2440 :
2450 DATA 32
2460 DATA Cmin,Cmin,G7,G7,G7,G7,Cmin
2470 DATA Cmin,Cmin,Cmin,G7,G7,G7,G7
2480 DATA Cmin,Cmin,Fmin,Fmin,Cmin,Cmin
2490 DATA G7,G7,Cmin,Cmin,Fmin,Fmin
2500 DATA Cmin,Cmin,G7,G7,Cmin,Cmin
2510 :
2520 DATA 16
2530 DATA Amin7,D7,Gmaj6,Emin6,Gmin9,C7
2540 DATA Fmaj7,Dmin6,Fmin7,G#min6
2550 DATA Gmaj6,D#M,Cmin6,D7,Bmin,E7
```

Le programme vous demandera en premier lieu le nombre de temps par mesure, le tempo et si vous voulez ou non une variation rythmique. La composition est basée sur une séquence d'accords résidant dans une instruction DATA, et le rythme peut être changé ou ajusté dans le programme.

Ce programme produit un volume sonore beaucoup plus fort que le programme du dernier chapitre et il vous faudra sûrement diminuer le volume pour éviter les distorsions.

Commentaires

La variable **registredaccords$**, en 1070, contient les accords disponibles, de la même manière que **gamme$** contient les notes disponibles. Le programme est approvisionné avec les accords suivants:

```
    M :  Majeur
    7 :  7ème de dominante
    9 :  Majeur, Neuvième
  min :  Mineur
 min6 :  Mineur, Sixte
 min7 :  Mineur, Septième
 min9 :  Mineur, Neuvième
 maj6 :  Majeur, Sixte
 maj7 :  Septième Majeur
  aug :  Augmenté
  dim :  Diminué
```

Les données des accords sont listées dans des instructions DATA, à partir de la ligne 2160. Le chapitre 2 nous donne quelques brèves informations sur les accords et la figure 2.11 illustre quelques-uns des accords les plus communs.

Les instructions DATA renferment uniquement la liste des notes qui rentrent dans la composition de l'accord, elles ne nous donnent aucune information sur la construction de l'accord. Le nombre de notes a volontairement été limité à six, extraites des octaves de 0 à 2.

Les notes sont arrangées de manière à souligner l'élément dominant de l'accord. Par exemple, dans un accord mineur avec une neuvième l'élément important est la neuvième et les notes de la ligne 2280 ont deux neuvièmes (notes D, ou RE, dans le cas d'une clé de C, ou DO). Plus

l'accord est complexe, plus on utilisera de notes de la gamme. Les notes des accords sont lues dans **note$** des lignes 1200 à 1250.

La mélodie est extraite des séquences d'accords contenues dans les lignes DATA démarrant en 2390. La première partie représente le nombre d'accords, qui est stocké dans la variable **noofchords**, en ligne 1130. Le reste des données représentent les accords qui sont lus et stockés dans la variable **melody$** en 1170.

La réponse à la question de la ligne 1340 est stockée dans la variable **noofbeats** et représente le nombre de temps par mesure. **Noofbeats** est utilisé en 1510 pour produire une composition basée autour de croches. Par exemple, une entrée de 4 produira des mesures à 8 croches.

Le tempo est demandé à la ligne 1350 et stocké en **tempo1**. Les altérations de rythme sont produites par une modification de cette entrée, **tempo1** est donc sauvé dans tempo.

La ligne 1380 vous demande si vous voulez ou non des variations de rythme. La variable **sync** (pour syncopation) est mise à -1 (vrai) ou 0 (faux) suivant votre réponse.

La boucle de 1430 à 1560 contrôle la génération de l'air de musique. Chaque accord dure le temps de la mesure. Une mesure est contituée de croches égales au nombre de temps dans une mesure (**noofbeats**), multiplié par deux.

Si une variation de rythme a été selectionnée, la ligne 1520 fait appel à la routine de variation. En 1500 nous sélectionnons de manière aléatoire une des quatre variations qui dure au maximum le temps d'une mesure, à moins qu'elle ne soit sélectionnée une seconde fois.

La routine "jouer" en 1610 est appelée par l'instruction se trouvant en 1530, et joue trois

notes. Cette routine effectue plusieurs travaux et si la valeur tempo est inférieure à 25 le programme Basic aura du mal à suivre, ce qui a pour effet de donner des résultats hésitants ou inégaux.

La routine "jouer" effectue un appel à une autre routine en 1740 pour analyser l'accord. L'analyse de l'accord se fait de manière identique à la routine d'analyse des notes. La première vérification consiste à examiner les deux ou trois premières lettres pour trouver la clef. Les autres lettres sont considérées comme étant le type de l'accord. Le calcul de la clef est effectué de la même manière que le calcul de la note (1760). En 1770 nous obtenons un nombre qui nous permettra de prendre deux notes différentes dans l'accord dans les lignes 1790 à 1840 et de terminer la routine.

La routine "Choisir une note" en 1880 est identique à la routine d'analyse de la note, mais elle est appelée deux fois, une fois pour chaque note que nous venons de tirer de l'accord. Les lignes 1620 a 1650 envoient les deux notes aux canaux A et C du générateur de son. L'acte final de la routine "jouer" est de produire une note de basse de la clef de l'accord. Ceci maintient juste la note de base de l'accord, par exemple un accord DO va produire une note basse de DO, et SOL une note basse de SOL . Les trois notes sont synchronisées (rendez-vous) pour être émises en même temps. En stéréo, ceci produit deux notes de part et d'autre avec une note de basse au centre.

La dernière routine produit les variations de rythme, elle est située en ligne 1970. Elle n'est appelée que si une variation a été demandée. Si oui, elle est appelée avant chaque note pour déterminer sa durée. Les variations sont produites par une altération de la valeur de la variable **tempo** en accord avec les temps. Cinq possibilités sont offertes en ligne 1970.

Essayons notre programme

Il y a deux solutions pour changer la sortie, en dehors de la variation de tempo et des temps par mesure. La première passe au travers d'une sélection d'accords utilisés dans les airs , la seconde implique la complexité des variations de rythme que vous introduisez.

Les données des lignes 2390 à 2400 sont basées sur le fameux rythme de blues à douze mesures. Le second jeu de DATA est composé de quelques accords neuvièmes qui donnent plutôt naissance à une ambiance un peu bruyante. Les DATA de la ligne 2450 produisent une séquence d'accords similaires à ceux que l'on peut rencontrer dans la musique électronique et à ce niveau vous préférez sûrement la musique sans variation de rythme. Le dernier jeu de DATA en 2520 est plutôt indiqué pour un tempo slow. L'utilisation d'accords neuvièmes mineurs, de sixte mineurs et de septièmes majeurs incite plutôt au style jazz.

Comme la sortie est déterminée par les accords que vous introduisez dans les instructions DATA, vous avez un contrôle complet de la progression des accords et vous pouvez essayer n'importe quelle séquence d'accords. Vous pouvez étendre la gamme des accords en incluant de nouvelles données pour les accords. Insérez un nouveau nom d'accord dans **registredaccords$** en 1070, augmentez **note$** en 1090 ainsi que le n de la boucle en 1210. Veillez à l'endroit où vous introduisez le nouveau label de l'accord : dans **registredaccords$** par exemple, si **min6** arrive avant **min** le programme nous donne un numéro d'accord incorrect pour **min** en ligne 1770.

Par convention un accord majeur est représenté par M. Dans la notation standard, un accord majeur est représenté tout simplement par lui-même comme DO ou C . Dans notre propre notation ceci se transformera en CM pour un DO majeur et en G M pour un SOL majeur. En

donnant un symbole à chaque accord nous simplifions la routine d'analyse des accords.

Pour augmenter les variations il est possible d'augmenter les notes attribuées à chaque accord. Ceci peut se faire simplement par une extension des notes sur plus d'octaves. Le fait d'augmenter le nombre de notes nous permet également d'introduire des accords plus complexes contenant plus de six notes. Vous pouvez l'expérimenter par l'exemple suivant. Un DO majeur est formé d'une gamme normale de DO. Un DO7 ou C7 est constitué d'une gamme de FA ou F majeure s'il contient un bémol et nous sommes dans une clé de FA. DO mineur contiendra un bémol de RE (MI bémol) gamme majeure.

Comme le programme prend aléatoirement des notes de l'accord, le fait d'utiliser des gammes complètes donnera naissance à des accords dissonants. Le principe de prendre des notes de la gamme appropriée est une autre manière de produire des airs de musique et nous verrons quelques suggestions un peu plus loin dans ce chapitre.

Une valeur de quatre temps produit une mesure de 4/4 et chaque accord sera tenu jusqu'à la fin de la mesure. Avec une valeur de 2 il semblera que la note ne sera tenue que jusqu'à la moitié de la mesure, etc . En fait, la seule chose que nous faisons est de diminuer le nombre de croches dans une mesure. Vous pouvez arranger l'enveloppe pour que le premier temps soit d'un volume plus important que les autres.

Ajoutez des variations de rythme

Ajouter des variations de rythme est une partie importante du programme, elle permet de supprimer la monotonie d'une séquence de croches. La méthode utilisée peut s'adapter à de multiples variations de rythme. Un coup d'oeil au premier exemple, en ligne 1990, nous montrera le principe.

Conformément à certains critères que vous avez programmés, entre autres le nombre de notes dans un temps, la durée de la note est altérée. A moins que vous ne désiriez une désynchronisation complète vous devez vous assurer que tout nouveau temps donné à une note est pris sur une autre. En tout cas, méfiez-vous: un ou deux contretemps ajoutent du piquant, beaucoup de contretemps désorganisent. 2120 nous montre ce qui peut arriver si nous ne suivons pas cette règle.

Si vous recherchez d'autres exemples dans la routine vous pouvez voir le principe de fonctionnement. Généralement, le temps est pris sur une note pour être restitué à une autre afin que chaque mesure contienne huit notes. Ce principe peut être adapté pour produire de nombreuses autres configurations de rythme.

La routine en 2050 est un peu différente en ce qu'elle ajoute des notes à la mesure. Ceci se fait par l'appel de la routine d'analyse des accords qui ajoute des notes d'une durée égale à la moitié du tempo.

Les variations des lignes 2080 et 2120 ne sont pas utilisées, comme nous l'indique le listing, mais, en ôtant les instructions REM elles peuvent facilement être incluses dans le programme et appelées par la ligne 1970. En changeant cela, ainsi que le pointeur **sync** en 1500, le programme aura une sortie complètement modifiée. Vous trouverez que les tempo en dessous de 25 ne pourront pas répondre aux variations; ceci est dû au langage interprété du Basic et de l'énorme travail pour calculer chaque note.

Autres contrôles pour une sélection aléatoire des notes

Comme vous l'avez réalisé, le programme émet une mélodie duophonique avec une troisième voix pour la basse. La basse a été introduite pour renforcer la tonalité de l'accord et générer un peu de volume.

Les deux notes de la mélodie ont la même durée, mais nous pouvons modifier cela pour que les notes empruntées à l'accord aient une durée différente. L'effet obtenu se rapprochera d'un effet polyphonique. Il est possible également d'intervenir sur la durée de la note basse du canal central.

Les routines de composition sont très complexes et un langage de programmation du style du Basic a du mal à suivre; la composition de pièces rapides est à exclure. Un exercice intéressant pourrait consister à mettre toutes les données intéressantes dans des **array** et en effectuer la lecture pour que la seule occupation du Basic consiste à effectuer les calculs. Une autre alternative pourrait être de stocker hauteurs et durées, et par la suite effectuer uniquement la lecture et l'émission sur un tempo quelconque. Ces "recettes" sont citées pour vos expériences futures.

Il existe d'autres manières pour choisir les notes d'une gamme, liées à un accord particulier. En fait, il y a même une multitude d'autres méthodes pour produire de la "musique informatique" et il faudrait écrire plusieurs autres livres pour les aborder toutes. Le but principal du présent livre n'étant pas la composition de musique sur ordinateur, alors que nous avons sous la main un programme pour réaliser des expériences, nous n'aborderons que les principes d'une autre méthode. Vous étudierez les détails tout seul, le programme n'étant guère plus compliqué que celui que nous venons de lister.

Autres extensions et modifications possibles

Dans ce chapitre ainsi que dans le chapitre 10 notre seul objectif était de contrôler l'élément aléatoire dans la musique d'ordinateur. L'élément aléatoire est obligatoire pour donner naissance à de la musique originale,

encore faut-il le contrôler et l'ajuster pour produire le résultat voulu.

Si vous introduisez des gammes complètes dans les instructions DATA du programme 11.1 vous n'obtiendrez probablement pas le même résultat que lorsque vous vous contentiez de six notes. Ceci est dépendant de la probabilité de sélection qui est identique pour chaque note. Si vous introduisez huit notes, la probabilité d'appel d'une note est de 1 sur 8. Ceci porte atteinte à l'effet mélodique et à l'effet harmonique.

Premièrement, les notes choisies sont trop aléatoires dans un cadre harmonique donné. Par exemple, dans la plupart des compositions "humaines" les mélodies basées sur des accords de DO mineur contiennent probablement plus de DO,SOL et MI bémols que d'autres notes (ceci n'est pas toujours vérifié mais nous nous servirons de cet exemple comme point de départ). Or, dans notre programme, nous ne faisons aucune discrimination dans la sélection des notes. Nous allons donc tout de suite modifier ces probabilités en dressant un tableau - disons que dans notre exemple les notes auront des "poids" différents suivant les accords.

Pour simplifier nous pouvons ignorer tous les bémols et les dièses. Une table de probabilité pour un accord DO mineur pourrait avoir la composition suivante:

```
DO0   = 20%
RE0   =  5%
RE 0  = 15%
FA0   = 10%
SOL0  = 15%
SOL 0 = 10%
LA 0  =  5%
DO1   = 20%
```

Ces tableaux sont basés sur des règles empiriques et peuvent être étendus facilement pour couvrir plus d'une octave. De même vous

pouvez inclure un SI. L'implantation de telles règles améliore sensiblement la sortie mélodique mais si vous élargissez votre gamme de notes à plus d'une octave les intervalles entre notes risquent d'être trop grands et nous nous éloignons un peu des règles de la composition classique. Mais là aussi, nous pouvons prévenir ce défaut en appliquant d'autres restrictions dans les probabilités.

Améliorons la mélodie

Si nous observons la partie mélodique d'une pièce de musique, nous pouvons nous apercevoir que l'intervalle des notes est toujours plus ou moins inférieur à la quinte (là encore nous faisons une généralisation pour une grande partie de la musique classique, pour le jazz et un grand nombre d'airs populaires). Pour améliorer notre programme, il faudra limiter les intervalles et intervenir en dressant un tableau des probabilités un peu similaire à celui du choix des notes.

Si nous classons les intervalles par leur fréquence d'apparition, les sauts d'un intervalle arrivent en tête, suivis des sauts de deux intervalles, puis de trois. Nous éviterons de tenir compte des bémols et dièses pour ne pas compliquer l'affaire. Notre table de probabilité aura la forme suivante:

```
+1 = 30%
-1 = 30%
+2 = 15%
-2 = 15%
+3 = 5%
-3 = 5%
```

Ce tableau, à nouveau, va nous servir de point de départ dans notre exposé et vous pouvez le modifier à votre gré. Si vous tenez compte de ces deux règles pour votre choix aléatoire, vous pourrez observer une très nette amélioration de votre mélodie.

Nous n'avons toujours pas abordé la partie concernant la durée de la note. C'est elle qui est la plus complexe à programmer dans la musique de composition. Dans un tel programme, les meilleurs résultats sont obtenus par un jeu pré-programmé de durées de notes ou simplement une suite de croches. Ceci produira probablement une interprétation de type BACH.

Il est évident qu'un tel programme sera considérablement amélioré si nous utilisons les trois canaux. Pendant que les deux canaux mélodiques poursuivent leur propre jeu avec une certaine indépendance l'un de l'autre, à part pour les sorties d'accords, vous trouverez sûrement que des notes qui avaient tendance à se heurter avant semblent passer les unes sur les autres; par exemple, un canal peut produire une mélodie descendante pendant qu'il poursuit son cheminement dans une gamme ascendante. Vous pouvez réduire le deuxième canal à l'émission des notes nécessaires aux accords et au support de la basse.

Contrôlons les notes de basse

Les notes de basse peuvent être contrôlées de deux manières. Premièrement, leur durée peut être fixée de manière à en jouer quatre dans une mesure, par exemple quatre noires, ou trois si le morceau est mesuré en 3/4, et deuxièmement vous pouvez restreindre le choix des notes, quintes et tierces par exemple, mais il vaut mieux faire des expériences.

Créons et développons nos propres programmes

Dans le cadre de toutes les règles et limites que nous venons de définir, vous avez la possibilité d'appliquer vos propres idées. Il faut néanmoins faire attention dans le choix des notes pour ne pas créer d' ambiguïtés dans les tableaux contenant les pourcentages de choix. Les routines de sélection des notes doivent prendre en considération toutes les procédures. Comme pour tous les programmes d'ordinateurs,

nous pouvons améliorer et développer notre programme de composition et vous trouverez ci-dessous quelques idées supplémentaires pour poursuivre vos investigations. Elles sont exprimées simplement en tant que suggestions, car s'il fallait entrer dans les détails des instructions et des listings, il faudrait un volume supplémentaire.

Il serait intéressant de permettre à l'opérateur ou au programmeur de changer les paramètres pendant que le programme tourne. Ceci peut être réalisé par des entrées au clavier ou alors de manière beaucoup plus subjective en permettant un contrôle des paramètres au travers d'une manette de jeu. Les possibilités sont fantastiques !

Le guide d'utilisation vous fournit tous les détails sur le contrôle des manettes de jeu et sur la méthode de récupération des résultats. Les paramètres que vous pouvez contrôler sont le tempo, la variation de rythme, les clefs ainsi que les accords. Décrivons une des méthodes possibles pour altérer les accords.

Il y a un cycle de progression des accords connu sous le nom de cercle des quintes et illustré en figure 11.1. Les accords septième, par exemple DO7, ont une tendance à aller vers des accords de quinte dans le bas de la gamme. Ainsi un DO7 aura tendance à aller vers un FA, un FA7 vers un LA ou SI bémol, qui lui-même aura tendance à aller vers un RE (Eb) etc. Le mouvement semble satisfaisant et si vous n'avez pas d'instrument de musique pour le prouver, l'AMSTRAD peut vous confirmer l'harmonie de ce cycle. Le cercle des quintes est très utile dans la composition, il nous montre la tendance de progression des accords.

Entre le mouvement circulaire d'une manette de jeu et le cercle des quintes, il n'y a qu'un pas (de programmation) à franchir. Il faut chercher à accéder aux accords par les mouvements de la manette, par exemple un tour à

droite nous permet de passer du DO au FA, puis au LA etc., non pas nécessairement pour avoir un accord rigoureux, mais plutôt pour obtenir une progression harmonieuse.

Vous désirez peut-être sauvegarder le résultat de votre composition, ou plutôt celui de l'AMSTRAD. Tous les programmes listés jusqu'à ce point, à part le programme 10.2, sont étudiés pour jouer une composition continue, composée instantanément dès la mise en route du programme. Il vous est possible d'ajouter quelques instructions qui, après chaque mesure, ou à un autre moment si vous préférez, vous posent la question: "Voulez-vous sauvegarder votre composition, oui ou non ?". Vous pouvez même songer à créer une bibliothèque et un catalogue des meilleures compositions de l'AMSTRAD. Consultez votre guide d'utilisation pour vous familiariser avec le maniement des fichiers.

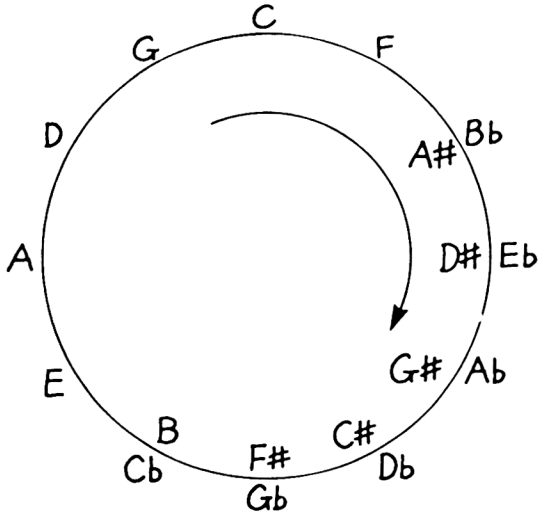


Figure 11.1.

Chahutons nos compositions

Le programme d'analyse des notes, au chapitre 10, produit des sonorités en fonction

de la fréquence de chaque note du morceau de musique. Il est possible d'utiliser ce même air comme base de départ et de produire une sortie complètement différente. Une des méthodes pourrait consister en une opération mathématique sur les hauteurs représentant les notes.

Les ordinateurs constituent un excellent moyen pour réarranger les relations mathématiques qui lient les notes entre elles ainsi que les gammes. Vous pouvez appliquer tout un jeu de fonctions mathématiques aux notes, mais nous allons juste nous contenter d'un seul exemple dans cet ouvrage. Le programme de composition "verlan" ne fait qu'inverser la hauteur des notes pour que les notes les plus basses soient jouées en aigu et inversement. Les résultats sont intéressants et très souvent amusants. Si nous changeons de fonction mathématique ou que nous permutons différemment, le résultat sera autre.

Comme nous sommes en possession de programmes nous permettant de jouer le Rondo de MOZART, la Marche de la Liberté ou la Danse de la Fée Prune (reportez-vous au chapitre 9), nous allons essayer notre nouvelle recette sur ces compositions. La première étape consiste à trouver la note la plus haute et la note la plus basse puis d'en déduire la note centrale pour déterminer le centre de la permutation. Par exemple, si la note la plus haute et la plus basse sont respectivement DO2 et SOL -1, la permutation se fera autour de LA 0, qui se trouve à égale distance des deux notes. Ajoutez les lignes suivantes à l'une des listes du programme 9.2

```
 100 REM PROGRAM 11.2
 110 REM Routine à l'envers
 120 REM Ajouter au Program 9.2
 130 :
1025 hipit=16:lopit=3822
1026 pp=349.228:REM Voir le texte
1843 GOTO 1850
1844 IF freq<pp+2 AND freq>pp-2 THEN
     1850
```

```
1845 semi=1
1846 IF freq>pp THEN GOSUB 1862:GOTO
     1850
1847 IF freq<pp THEN GOSUB 1866:GOTO
     1850
1851 IF hauteur>hipit THEN hipit=hauteur
1852 IF hauteur<lopit THEN lopit=hauteur
1861 :
1862 freq=freq/2^(1/12)
1863 IF freq<pp+2 AND freq>pp-2 THEN
     freq=freq/2^(semi/12):RETURN
1864 semi=semi+1:GOTO 1862
1865 :
1866 freq=2^(1/12)*freq
1867 IF freq<pp+2 AND freq>pp-2 THEN
     freq=2^(semi/12)*freq:RETURN
1868 semi=semi+1:GOTO 1866
```

Commentaires

La première étape consiste à chercher la note la plus haute et la note la plus basse. L'ordinateur peut se charger de ce travail. Après l'exécution du programme modifié, faites un PRINT des variables hipit et lopit. Un coup d'oeil au tableau des notes représenté dans l'annexe 1 vous permet de déterminer rapidement la note se trouvant à mi-chemin. Vous n'êtes pas obligé de déterminer la note centrale avec une extrême précision, mais vous vous apercevrez qu'elle convient bien mieux dans la plupart des cas. Si les notes produites par la nouvelle routine s'égarent de trop, cela provient de la génération de hauteurs aberrantes, hors-limites.

Après avoir choisi la note centrale, il faut également assigner une fréquence au point de pivot, pp, en ligne 1026. Puis ôtez la REM de la ligne 1843.

Pour la version "huit mesures" du Rondo, la valeur du point de pivot est: 349.228 ou 369.994. Les valeurs se transforment en 329.628 ou 349.229 pour la version complète. Pour la Marche de la Liberté, la valeur est de 415.305.

Elle est de 622.254 pour La Fée Prune.

Le programme comporte deux routines de sélection de notes suivant que la note se trouve au-dessus ou en-dessous du point de pivot. Elles sont situées en 1862 et 1866 et travaillent de manière similaire. Les formules en 1862 ou 1866 incrémentent ou décrémentent les fréquences par bonds de demi-tons, elles ressemblent aux formules que nous avons décrites au chapitre 4. Les instructions en ligne 1863 et 1867 comparent la nouvelle fréquence par rapport au point de pivot, nous permettant une variation de plus ou moins 2, la formule ne pouvant nous délivrer une fréquence exacte. Si vous avez des problèmes d'interprétation, vérifiez en premier lieu si cette limite de 2 convient. La variable **semi** comptabilise l'écart de la note par rapport au point de pivot, en demi-tons. Cette variable est utilisée par la suite en 1863 et 1867 pour calculer la nouvelle fréquence, en utilisant la même formule.

Le programme sera nettement ralenti par l'exécution de tous ces calculs.

Nous ne touchons pas à la durée des notes et c'est ce qui permet une certaine reconnaissance des interprétations, mais il est possible de leur appliquer une fonction similaire pour embrouiller un peu plus votre auditoire.

L'exemple que nous venons de voir est très simple, vous pouvez vous amuser en appliquant d'autres fonctions comme SIN et COS à la place de la permutation simple. De même, vous pouvez changer l'incrément d'un demi-ton en d'autres valeurs. Pour plus de détails, reportez-vous au chapitre 4, au paragraphe traitant des gammes microtonales.

La progression verlan (à l'envers)

Il est facile de changer l'ordre de sortie des notes contenues dans une variable "array", nous pouvons même les faire jouer dans un ordre

inverse. Replacez les lignes suivantes dans un des programmes dérivés du programme 9.2

```
  10 REM PROGRAM 11.3
  20 REM Air à l'envers
  30 REM Changer ces lignes dans
     PROGRAM 9.2
  40 REM Aussi rendezvous
  50 REM des dernières notes-Voir texte
  60 :
1660 voca=va+1:vocb=vb+1:vocc=vc+1
1700 WHILE voca>0 OR vocb>0 OR vocc>0:
     WEND
1890 voca=voca-1:IF voca<1 THEN RETURN
1950 vocb=vocb-1:IF vocb<1 THEN RETURN
2010 vocc=vocc-1:IF vocc<1 THEN RETURN
```

Commentaires

Les commentaires vont être très simples: au lieu de lire l'**array** de façon normale, nous faisons une lecture à partir de la dernière note. Il faut vous assurer que la dernière note de chaque canal soit affectée d'un rendez-vous pour commencer l'exécution en synchrone. Vous pouvez ajouter une fausse "dernière note", par exemple un silence, à chaque canal, puis ajouter 1 à **va**, **vb** et **vc** ou ajouter physiquement un rendez-vous aux données des notes. Les dernières notes de chaque canal ne se présentent pas toujours au même instant; assurez-vous de bien synchroniser sur la note qui convient. Dans la version première du programme 9.2, les additions seront faites de la manière suivante:

```
2150 DATA $38,E1,12
2260 DATA $38,E-1,12
2370 DATA $38,R,12
```

Utilisez également des enveloppes inverses. Combinez les effets des deux derniers programmes et changez le tempo des interprétations. Essayez également de jouer les notes de manière aléatoire.

Dans le "show-biz", si un chanteur est

arrivé au "top-niveau" il est souvent de règle de sortir immédiatement une deuxième chanson pour tirer avantage de ce succès. Très souvent cette seconde chanson est une mixture de la chanson à succès, utilisant les mêmes progressions d'accords et les mêmes mouvements mélodiques, et il est peu probable que les auteurs utilisent des ordinateurs. Vous êtes donc devant de nombreux hits et des chansons à succès qui doivent suivre...

La chanson automatique

Il existe maintenant des logiciels simulant la voix humaine; on peut les programmer à une certaine hauteur pour produire des chansons ou des chants. Si vous arrivez à lier un programme de composition et un programme de génération de poésie, vous aurez sûrement une grande audience chez les auteurs-compositeurs. Même si cette technologie n'est pas encore disponible sur l'AMSTRAD à ce jour, il est toujours possible de lier un programme de musique à un programme de génération de poésie. Nous pouvons démarrer par un programme qui compte le nombre de syllabes dans un vers et qui compose de la musique avec le même nombre de notes. Si vous avez la chance de posséder un système de génération de la parole, vous pouvez vous amuser à composer la musique de fond pendant que le système superpose le texte, un peu dans le style de Léonard COHEN.

Ces suggestions sont faites pour vous aider à vous lancer. La musique de composition informatique est un domaine tout nouveau qui est encore très peu exploré à ce jour. S'il vous tente, vous pouvez y exercer vos talents.

Musique sur l'Amstrad, demain

Comme vous le savez, l'AMSTRAD peut être considérablement étendu. L'ajout de lecteurs de disquettes, d'imprimante et de ROM supplémentaire augmentent considérablement le

potentiel et la puissance de votre machine.

Les nouvelles technologies résoudront bien des problèmes, relatifs non seulement à l'ordinateur mais aussi à la musique. Si les constructeurs prennent en considération les facilités d'adaptation et d'extension de l'AMSTRAD, nous pouvons espérer voir un jour des claviers du type piano et des modules pour compléter la fonction synthétiseur.

Les micro-ordinateurs, par l'amélioration incessante de leurs performances et de leur souplesse d'utilisation, s'implantent de plus en plus dans le domaine de la génération du son, ainsi que dans le domaine de l'enregistrement et du playback. Beaucoup de studios d'enregistrement sont équipés d'ordinateurs, et la tendance consiste à accroître l'équipement. Votre futur micro-ordinateur comportera un jour des fonctions semblables à celles des ordinateurs de studios.

Dans ce monde où le développement de la technologie est d'une rapidité fulgurante, qui pourra préjuger de quoi seront faits les lendemains de l'AMSTRAD ?

Annexe 1
Les notes, leur fréquence et leur hauteur

Dans le tableau suivant, la numérotation des octaves est identique à celle du guide d'utilisation. La numérotation entre parenthèses indique un repérage différent, expliqué dans le Chapitre 4 de ce livre.

Les fréquences affichées sont celles des notes utilisées dans les gammes occidentales. Les hauteurs sont déterminées par calcul, elles ne correspondent pas toujours exactement aux fréquences, mais la différence est si faible qu'elle est inaudible. Les erreurs relatives sont listées dans l'annexe 7 du guide d'utilisation.

Le DO ou C du milieu étant dans l'octave 0, il est noté comme C0.

NUMERO DE LA NOTE	NOTE	FREQUENCE EN HERTZ	HAUTEUR
	OCTAVE -3 (1)		
1	C-3	32.703	3822
2	C#-3	34.648	3608
3	D-3	36.708	3405
4	D#-3	38.891	3214
5	E-3	41.203	3034
6	F-3	43.654	2863
7	F#-3	46.249	2703
8	G-3	48.999	2551
9	G#-3	51.913	2408

10	A-3	55.000	2273
11	A#-3	58.270	2145
12	B-3	61.735	2025
	OCTAVE -2 (2)		
13	C-2	65.406	1911
14	C#-2	69.296	1804
15	D-2	73.416	1703
16	D#-2	77.782	1607
17	E-2	82.407	1517
18	F-2	87.307	1432
19	F#-2	92.499	1351
20	G-2	97.999	1276
21	G#-2	103.826	1204
22	A-2	110.000	1136
23	A#-2	116.541	1073
24	B-2	123.471	1012
	OCTAVE -1 (3)		
25	C-1	130.813	956
26	C#-1	138.591	902
27	D-1	146.832	851
28	D#-1	155.564	804
29	E-1	164.814	758
30	F-1	174.614	716
31	F#-1	184.997	676
32	G-1	195.998	638
33	G#-1	207.652	602
34	A-1	220.000	568
35	A#-1	233.082	536
36	B-1	246.942	506
	OCTAVE 0 (4)		
37	C0	261.626	478
38	C#0	277.183	451
39	D0	293.665	426
40	D#0	311.127	402
41	E0	329.628	379
42	F0	349.228	358
43	F#0	369.994	338
44	G0	391.995	319
45	G#0	415.305	301
46	A0	440.000	284
47	A#0	466.164	268

48	B0	493.883	253
	OCTAVE 1 (5)		
49	C1	523.251	239
50	C#1	554.365	225
51	D1	587.330	213
52	D#1	622.254	201
53	E1	659.255	190
54	F1	698.457	179
55	F#1	739.989	169
56	G1	783.991	159
57	G#1	830.609	150
58	A1	880.000	142
59	A#1	932.328	134
60	B1	987.767	127
	OCTAVE 2 (6)		
61	C2	1046.502	119
62	C#2	1108.731	113
63	D2	1174.659	106
64	D#2	1244.508	100
65	E2	1318.510	95
66	F2	1396.913	89
67	F#2	1479.978	84
68	G2	1567.982	80
69	G#2	1661.219	75
70	A2	1760.000	71
71	A#2	1864.655	67
72	B2	1975.533	63
	OCTAVE 3 (7)		
73	C3	2093.004	60
74	C#3	2217.461	56
75	D3	2349.318	53
76	D#3	2489.016	50
77	E3	2637.021	47
78	F3	2793.826	45
79	F#3	2959.955	42
80	G3	3135.963	40
81	G#3	3322.438	38
82	A3	3520.000	36
83	A#3	3729.310	34
84	B3	3951.066	32

OCTAVE 4 (8)			
85	C4	4186.009	30
86	C#4	4434.922	28
87	D4	4698.636	27
88	D#4	4978.032	25
89	E4	5274.041	24
90	F4	5587.652	22
91	F#4	5919.911	21
92	G4	6271.927	20
93	G#4	6644.875	19
94	A4	7040.000	18
95	A#4	7458.621	17
96	B4	7902.133	16

Annexe 2
L'entrée, la protection et l'exécution des programmes

L'incident le plus malheureux qui puisse arriver à un programmeur est de perdre le programme qu'il vient péniblement de taper et de mettre au point. Ceci arrive à tout programmeur au moins une fois dans sa vie. Les raisons de cette perte peuvent être très variées:

1) Vous avez pu tout simplement arrêter votre ordinateur sans faire une sauvegarde de votre programme.

2) Vous avez peut-être sauvegardé votre programme sur une seule disquette ou cassette, et vous n'arrivez plus à la relire.

3) Quelqu'un, par inadvertance, peut débrancher votre ordinateur.

4) Peut-être avez-vous exécuté votre programme avant d'avoir réalisé une sauvegarde, et voilà que votre programme se "plante".

La plupart de ces problèmes peuvent être évités, il suffit d'être aussi logique et méthodique que votre ordinateur ! Si cela ne vous est jamais arrivé, j'espère qu'il en sera ainsi pendant de longues années encore. Voici quelques suggestions pour prévenir tout incident fâcheux:

1) Vérifiez les câbles: risquent-ils d'être débranchés par un mouvement malencontreux ?

2) Imposez-vous une sauvegarde avant l'exécution de chaque programme, et si vous avez assez de patience, faites CAT. Cela est assez laborieux avec un lecteur de cassette, mais, même si 99 fois sur 100 vous n'avez pas de problème, il se peut qu'à la centième les choses se passent mal.

LES CASSETTES

Si vous utilisez des cassettes, une bonne précaution pour réaliser des enregistrements sans problèmes est de nettoyer régulièrement les têtes de l'appareil. Les cassettes bon marché ont une fâcheuse tendance à encrasser leur tête avec leur propre vernis. Evitez l'utilisation de cassettes auto- nettoyantes. Il est préférable d'appliquer, à l'aide d'un coton-tige, un produit de nettoyage que vous trouverez en vente dans les boutiques hi-fi.

L'ENTREE DES PROGRAMMES

Les programmes sont listés dans un format de 40 caractères par ligne - le nombre de caractères par ligne en MODE 1 . Ceci vous aidera si vous voulez vérifier vos programmes par rapport au listing.

Vous savez probablement qu'il est possible de programmer des touches de fonction pour vous aider lors de l'entrée du programme. Tous les détails vous sont donnés au chapitre 1 du guide d'utilisation, page 13 (version anglaise). Vous pouvez même essayer de développer vos propres fonctions en affectant aux touches des commandes que vous utilisez souvent: LOAD, LIST, CAT, AUTO...

Tout au long des programmes on a affecté les variables de noms représentatifs et tous les noms sont en minuscules. L'AMSTRAD ne convertit que le nom des commandes en majuscules, il est donc facile de s'y retrouver au moment de la

mise au point des programmes.

Nous avons également évité l'utilisation de techniques sophistiquées qui auraient rendu le programme trop difficile à suivre. Le but était de vous initier à la composition musicale sur l'AMSTRAD et de mettre à votre disposition des programmes simples et compréhensibles.

Il est toujours possible de réduire les programmes pour rendre l'exécution plus rapide, en utilisant des techniques de programmation un peu plus évoluées et en mettant plusieurs instructions sur une même ligne.

LA FUSION DES PROGRAMMES

Certains programmes de ce livre utilisent des routines identiques ou similaires, pour vous permettre de gagner du temps. Elles ont été construites de manière à pouvoir s'intégrer directement dans d'autres programmes. Ces programmes ne peuvent s'exécuter seuls. Par exemple le programme 5.5 doit être fusionné avec le programme 5.4.

Les programmes originaux sont tous numérotés de 10 en 10. Pour diverses raisons il n'a pas été possible de le faire dans les autres programmes. La plupart sont très courts et peuvent être ajoutés aux autres programmes; il suffit de charger le premier et d'entrer le second au clavier. Il est essentiel d'entrer chacune des lignes, y compris celles qui ne contiennent qu'une virgule.

D'autre part, on peut fusionner les programmes en utilisant la commande MERGE (voir le guide d'utilisation au Chapitre 8 page 27). Si on sauvegarde un programme isolé, on peut le fusionner avec plusieurs autres à l'aide de cette commande. Cette méthode vous servira également si vous utilisez des cassettes, comme indiqué dans l'introduction.

REINITIALISATION DU GENERATEUR DE NOMBRES ALEATOIRES

Dans vos différents essais, vous voudrez peut-être répéter une séquence de nombres aléatoires. Ceci peut se faire très facilement par une initialisation de la variable RND avec RANDOMIZE tel que:

```
RANDOMIZE 1
```

Là aussi, vous trouverez tous les détails dans le guide d'utilisation.

Surtout expérimentez, amusez-vous bien,... et en avant la musique!

LA BIBLIOTHÈQUE EDIMICRO

- *Collection « Ordinateurs professionnels »*

APPLE

de Merly ..... **GUIDE DE L'APPLE**
Tome 1 : l'Apple standard
Tome 2 : les extensions
Tome 3 : les applications

MACINTOSH

Gaucherand ..... **MACINTOSH : outils, progiciels, applications**

SINCLAIR QL

Tenin, Van Thong ..... **Guide pratique du Sinclair QL**
K et S Brain ..... **Intelligence artificielle sur QL**

- *Collection « Progiciels »*

APPLE

Bonnet, Dinh ..... **Multiplan sur Apple**

IBM PC

Bonnet, Dinh ..... **Lotus 1-2-3 à votre portée**
Bonnet, Dinh ..... **Multiplan sur IBM PC**
Commandeur ..... **Framework sur IBM PC : Exercices de gestion**

MACINTOSH

Bouilloux ..... **Multiplan et Chart sur Macintosh**

GENERAL

Bonnet, Dinh ..... **Mémento Multiplan**

- *Collection « Langages »*

Gaucherand, Lamoitier ..... **Fichiers en Basic par l'exemple**
Perbost, Berthet ..... **Introduction à MSX**

- *Collection « Ouvrages de base »*

Lamoitier ..... **Le Traducteur Micro**
Darnis, Van Thong ..... **Graphisme et CAO**
Avon ..... **Courbes de maths en Basic**

- *Collection « Ordinateurs familiaux »*

VG 5000

Amsler, Bardon ..... **Guide du VG 5000 Philips**
Amsler, Villemaud ..... **Jeux sur VG 5000 Philips**

MO5

Bieber, Perbost, Renucci ..... **Tout sur le MO5**
Perbost, Renucci ..... **Jeux sur MO5**

AMSTRAD

Penfold ..... **L'Amstrad avec plaisir**
Waugh ..... **Musique sur Amstrad CPC 464 et 664**

ATMOS/ORIC 1

Kosniowski ..... **Nouveaux Jeux sur ATMOS**
Chane-Hune, Darbois ..... **Guide de l'Oric**
Bayvejiel ..... **Jeux Graphiques sur ATMOS**
Viguier ..... **Premiers Pas en Programmation sur Oric**

SPECTRUM

Bridge, Carnell ..... **Aventures sur Spectrum**
Hurley ..... **Jeux Graphiques sur Spectrum**

COMMODORE 64

Fleurier, Meiller ..... **Jeux sur Commodore 64 : Jeux d'adresse et de hasard**
Fleurier, Meiller ..... **Jeux sur Commodore 64 : Jeux d'action et de réflexion**

MO5
ELECTRON/BBC

Bennani, Chaieb ..... **Graphisme et sons sur Electron et BBC**
Perbost, Renucci ..... **6 jeux : action et réflexion sur TO7**

PHILIPS C7420 VIDEOPAC +

Bardon, de Merly ..... **Jeux sur Philips C7420 Vidéopac +**

TO7

— ouvrages
Bieber, Perbost, Renucci ..... **Guide du TO7**
Perbost, Renucci ..... **Jeux sur TO7**
— logiciels sur cassette
Perbost, Renucci ..... **4 jeux avec manette sur TO7**
Perbost, Renucci ..... **6 jeux : action et réflexion sur TO7**

GENERAL

de Merly ..... **Ordinateur familial : Que choisir ?**
Djama ..... **Je construis mon premier robot**
Ly ..... **Applications familiales en Basic MSX**

Nombreux autres titres à paraître. Catalogue sur simple demande.

EXTRAITS DE PRESSE

GUIDE DE L'APPLE

« Un des meilleurs sur la place. »

Le Figaro.

« Le livre que nous attendions : complet, clair et pratique.

Apple France.

MULTIPLAN ET CHART SUR MACINTOSH

« Présente l'incomparable avantage de lier Microsoft Chart et Microsoft Multiplan... »

Bernard Vergnes, Directeur de Microsoft pour l'Europe.

MULTIPLAN SUR APPLE

« Nul doute qu'avec un tel outil vous gagnerez en temps et en efficacité. »

Microsoft France.

« Une bonne introduction... Un bon guide... Un bon investissement. »

Science et Vie Micro.

FICHIERS EN BASIC PAR L'EXEMPLE

« Cet ouvrage est recommandé aux personnes amenées à entreprendre la programmation des fichiers du Basic, pour lesquels il constitue une étude pratique, sérieuse et en profondeur. »

Ordi Magazine.

MULTIPLAN SUR IBM PC

« Ce livre rendra de nombreux services. »

OPC.

« Des modèles de qualité professionnelle. »

L'Ordinateur Personnel.

JEUX GRAPHIQUES SUR SPECTRUM

« Ce livre constitue une véritable bibliothèque de jeux variés. »

Plein Jeu.

JEUX SUR COMMODORE 64

« Cet ouvrage permet aux débutants de s'initier au langage Basic en s'amusant... Un livre d'un bon niveau. »

TILT

« Les programmes complets et pas trop longs utilisent bien les possibilités du C64. »

L'Ordinateur Individuel.

ORDINATEUR FAMILIAL : QUE CHOISIR ?

« Ce livre a le mérite de simplifier le choix dans la jungle des boutiques informatiques. »

Décision Informatique.

PREMIERS PAS EN PROGRAMMATION

« Les exemples sont clairs et bien choisis, la méthode simple et rationnelle. »

Microric.

« Une approche originale dans la littérature consacrée aux micros. »

Décision Informatique.

GUIDE DU TO7

« Ce guide a le privilège d'être présenté de façon claire, dans un style parfaitement accessible à tous. »

Micro 7.

« Un manuel de référence absolue. »

Le Figaro.

JEUX SUR TO7

« De nombreux conseils et « trucs » pour une programmation rapide et efficace. »

Décision Informatique.

« Un excellent livre d'approche, plein de renseignements utiles. »

Micro 7.

GUIDE PRATIQUE DU SINCLAIR QL

« Une somme impressionnante d'informations, des programmes nombreux et intéressants... Un bon outil d'initiation à la pratique de notre Sinclair QL. »

Direco International-Sinclair.

Nombreux autres titres à paraître. Catalogue sur simple demande.

EDIMICRO 121-127, avenue d'Italie, 75013 Paris

EDIMICRO 121-127, avenue d'Italie, 75013 Paris

Achevé d'imprimer en Septembre 1985
sur les presses de Jouve, 18, rue Saint-Denis, 75001 Paris
Dépôt légal : Septembre 1985 N° d'imprimeur : 14732

MUSIQUE SUR AMSTRAD CPC 464 ET 664

I. WAUGH

En avant la musique sur Amstrad : jazz, musique classique, effets spatiaux, chansons populaires, cris d'oiseaux... Avec ce livre, vous pourrez vous constituer une vraie bibliothèque de programmes musicaux. Bien mieux, vous deviendrez compositeur : vous rechercherez des harmonies, et vous ferez vous-même des transpositions musicales.
Tous les programmes sont en BASIC et sont accompagnés de commentaires clairs et précis : les débutants en informatique les comprendront aisément ; les musiciens « en herbe » y trouveront une initiation aux techniques élémentaires de la musique.

L'auteur

Ian WAUGH est musicien de profession. Passionné d'ordinateurs, il est l'auteur de la série « Faites de la musique sur BBC, Commodore 64, Amstrad... » qui a reçu le meilleur accueil de la presse britannique :

« Les meilleurs livres pour composer de la musique sur ordinateur..., des ouvrages instructifs et amusants..., un vrai tremplin pour aborder les applications musicales des microordinateurs... »

9 782904 457418

Edimicro
121-127, Avenue d'Italie 75013 Paris

ISBN : 2-904457-41-0

148 F

Edimicro

MUSIQUE SUR AMSTRAD

Ian Waugh

Document numérisé avec amour par

AMSTRAD CPC

MÉMOIRE ÉCRITE

https://acpc.me/